

YOUR 

MARCH 1985

£1.35

# COMMODORE

## COMMODORE GRAPHICS – THE INSIDE STORY

COVER WARRIOR:  
THUNDER BLADE  
ON EIGHT-BIT MODE  
AND GRAPHICS  
WITH  
TOWN ATTACK



UNPENTABLE PATTERNS

LETTER WRITER ▲ BASIC WORKBOOK ▲ FESTIVAL TIME





# CONTENTS

VOLUME 5  
NUMBER 6



The Simpsons



The Simpsons



The Simpsons

## FEATURES

- |   |   |
|---|---|
| • <b>Graphics Guide</b> 10                | • <b>Operation Wolf</b> 28                      |
| Get better graphics from your Commodore   | Arcade action                                   |
| • <b>Thunder Blade</b> 17                 | • <b>Palat</b> 36                               |
| You and your helicopter against the world | A programmer's language                         |
| • <b>Microprose Soccer</b> 20             | • <b>Tank Attack</b> 38                         |
| A game of two halves                      | An unusual new war game                         |
| • <b>Pac Mania</b> 25                     | • <b>Power Play Hockey</b> 43                   |
| Latest variation on the Pacman theme      | A tough new ice hockey game                     |
| • <b>TKO</b> 26                           | • <b>Introduction to Graphics on the C64</b> 45 |
| An exciting new boxing simulation         | Discover the graphics potential of the C64      |
| • <b>Risk</b> 27                          |   |
| A new version of the classic boardgame    |   |

## REGULARS

- |                                       |  |
|---------------------------------------|--|
| • <b>Data Statements</b> 6            | • <b>Relative File Programming</b> 50      |
| More news from the world of Commodore | Some more expert filing advice             |
| • <b>A Flow of Ideas</b> 21           | • <b>Letters</b> 51                        |
| More about flowcharts                 | Long live the C128                         |
| • <b>Software Spellbook</b> 23        | • <b>Readline Matters</b> 53               |
| The references of machine code        | Expanding your library of handy routines   |
| • <b>Extending Basic</b> 29           | • <b>First Steps</b> 54                    |
| Adding commands to your C64           | Error messages on the C16, C128 and Plus 4 |
| • <b>Games Update</b> 33              | • <b>Contributions</b> 55                  |
| The latest games for the C64 and 128  | How to write for Your Commodore            |
| • <b>Technical Troubles</b> 40        | • <b>Back Page</b> 90                      |
| Our technical experts start           | Corrections, info and 'The Nibbles'        |

## PROGRAMS

- |   |   |
|---|---|
| • <b>Letter Writer</b> 14                           | • <b>Personal File</b> 44               |
| Perfect printed letters without a wordprocessor     | Avoid time consuming editing procedures |
| • <b>Basic Workshop</b> 32                          | • <b>Electronic Notebook</b> 58         |
| A new program to help you avoid typing all keywords | A personal diary on disk                |
| • <b>Sprite Library</b> 42                          | • <b>Head for Home</b> 57               |
| More tips on sprite manipulation                    | Boardgame fun for the C16               |

**Editor/Sheet Lead:**  
**Deputy Editor:**  
**Paul Whittington**  
**Technical Editor:** Paul Fox  
**Advertisement Manager:**  
**Paul Aspinall**  
**Ad Copy Control:**  
**Andrew Schmitt**  
**Artist:** Alan Bouldier  
**Designer:** Neil Swaine  
**Composition:** Plover  
**Printers:** Linn Ltd

Your Commodore magazine appears on the first Friday of each month. Argus Specialist Publications Limited, Editorial & Administration Offices, Year Commodore, No 1 Gables, Queens, London W2E 6JL. Telephone: 01-837 9425 Telex: 8871488

Subscription rates upon application to Year Commodore Subscriptions Department, Richard Ltd, 1, Royal Park House, Brooklands, Woking, Surrey GU24 0HL, U.S.A. Subscriptions Agent: New-Old Worldwide Publications, 1311 West 25th Street, Toronto, Canada M6P 1C6 USA

**ARGUS  
PRESS  
GROUP**

The contents of this publication including all articles, designs, drawings and programs and all copyright and other intellectual property rights therein belong to Argus Specialist Publications Limited. All rights reserved by the Law of Copyright and other intellectual property rights and by virtue of international copyright conventions are specifically reserved to Argus Specialist Publications Limited and any reproduction requires the prior written consent of the Company. © 1989. Distributed by D.N. Distribution & Sales, 1000 Court Road, London W2E 6JL. Printed by Chase Web, Plymouth. Options requested to correct are the opinion of the reviewers and not necessarily those of the magazine. While every effort is made to thoroughly check programs published for errors we cannot be held responsible for any errors that do occur.

ISSN 0269-9277



# Data Statements



Radio 1 DJ, Simon Bates (right), uses the Commodore 64

## R1 DJ Meets CBM

**R**adio 1 disk jockey Simon Bates was given the works at the Commodore Christmas Show in more than one sense. Apart from being given VIP treatment from Commodore boss Steve Franklin, Bates was also presented with a copy of *The Works* by Bates and Wally Dutton's MD, Jan Hawkins.

Of special interest to the Argus-loving DJ was the many name catches Nigel Jones, Commodore's answer to Stock, Atkins and Waterman put the MIDI-linked computer through its paces.

What Bates caught some of the people on Radio 1 may be Argus instead in the future.

## Dabbers For Epson

**P**rinter manufacturers, Epson, have started the New Year by releasing all the way to Herod's Hospital from their Genshō offices.

The printing company's new address is Epson (UK) Ltd, Company 180, Marlow's Avenue, Royal Wootton Bassett, Herts HP12 2EZ. Tel: (0442) 81144.

## CBM Roadshow

Commodore are going to *Context* during the early part of this year as their education team has the word. An annual deal was made by holding an exhibition at Middlesex Polytechnic's Borena Green site and it proved that there was sufficient interest from

educationalists to give the event light to the roadshow.

All the polytechnics, the Amiga and PC were put through their paces and the new Aurora BBC emulator received a lot of interest as a possible bridge to the current machines possessed by schools.

Dates and venues for the roadshow have yet to be

announced but Peter Talbot, Commodore's National Sales Manager for Education, has high hopes. "I was delighted both with the turnout and the caliber of people attending our show. This is definitely going to become an annual event."

"Our 1988 education initiative has seen Commodore making major

strides on the education market, and 1989 will be even better."

**Touchline:** CBM (UK) Ltd, Commodore House, The Touchline Garage Road, Maidenhead, Berks SL6 7XA.

*Commodore's Place Exhibition at the Computer in Education Conference*



## Script Software

With a reputation for its high quality, to give the far east, from the Indian sub to the computer world, Scripta Visual finds that women get a raw deal when it comes to the advertising and content of computer games, and has formed the Organisation Against Sexism in Software to bring pressure to bear on the offending software houses.

The campaign has no real switching to ensure that a standard of moral values is maintained and the issue more has been made as

opposed to the unflattering portrayal of women. According to OASIS, women are mostly portrayed either as busy bimbos, or else weak creatures who must be rescued from perilous situations by male heroes. More than this, too much violence is practised to appeal to the male ego, leaving the misrepresentation of female participation.

Although computer gaming is still very much a male preserve, the number of female computer

enthusiasts is growing. Scripta does not expect special status to be given to women, merely an acceptance that they do exist, by avoidance of the standard male female stereotypes in advertising, and gender determinations in playable games.

OASIS already has an active membership but is open to women like Tim Duck or Harriet who sympathise with the cause. An annual subscription costs £3 and includes six copies of the organisation's bi-

monthly magazine.

One example of what's at stake is already pressed on the first issue of the OASIS magazine is *Palace Software's* *Seraphim* promotion which features the provocative character of Maria Whitehead. It is exploitation, or merely a steamy D-trap?

**Touchline:** OASIS, Scripta Visual, 3 Alden Court, South, Road, Wokingham, London RG40 1BQ.

### Tower of Power



**M**TC's business, it is, has moved to new MC facilities, but owners of Commodore computers shouldn't be too off to the

## Summit For Sustainability

**S**outhern California may be the competition's last summer, and the price of a collection of great condensation-free walk-in coolers was recently awarded to Alan Clark from Carlsbad.

The competitors were left in awe of the launch of the "sumo" label by Alternated Softwear boss Roger Butler. Based on the fiscal theme of the game *Sumo*, which was directed by Roger's father as the Thomas, the competitors had to convert the £3.99 price of *Sumo* into mass and L&L Dracemany Atlas was the winner of the first category, picking money from the hat and cashed in on the prize.

Two forthcoming releases from Samuel are *Johnnie Sweeney's* The Roadie (new Grains Gertie) and *Dumbass*. Also Eddie Sweeney's full-page comic label, *Spine*. Again has released a game based on last year's craze in the Arabian Gulf, *Operation Hormuz*, programmed by David only in 89 on the C64, but the Amiga version won't follow until later this year.

Foreign Born	Native Born
1970-1979	1970-1979
1980-1989	1980-1989
1990-1999	1990-1999
2000-2009	2000-2009
2010-2019	2010-2019
2020-2029	2020-2029

little because an 4-ampere rate of administration would have given us better results.

The trolley is supplied with a four-wire power board, and is designed so that it can be pulled over any standard height dock or lane. The range of fixtures available are light pole, lametric, downcast rail or an appealing, heavy chandelier.

**Teacher: MDC Education**  
**1145 East Parker Ave**  
**Edwards Road, Nevada**  
**Hesperus, New Mexico 87521**  
**505-764-6711**

**Figure 6**

Following a 1997 initiative, Government Technology UK plant visitors can push an 'Academy' into the top priority manufacturers in the sector.

Second Commodore owners may remember that the CBM 81C1520 printer was a truly designed Seikosha machine, but since that time and direct data, Seikosha has entered the new age of technology. Last year saw the release of the heavy duty 58P00 printer, which reached the truly speed of 600cps making it the latest dot matrix printer currently available.

1988 was the start of the market boom with the



1000

1000

**E**xcellent! Arts has published a series of detailed color books for the two growing ranges of televiewing adventures. Doubleday's *Hawaiian Islands Says and Shows Tale P & M* and *TV viewers can buy the books from E.A.C. Publications*. **SUPPLY:** *Shoreline*, *TV*, *Shoreline*, *TV*.

Franchiser: **Pharmacia** / **AAU**  
 Customers / Suppliers: **Large**  
 Business Cycle: **11-16**  
 Season: **None** / **Length:**  
 Single Units: **50-115** / **Ref:**  
**00000000000000000000**

Intel's 486 133 pin package is offered as the alternative to the 486 technology favoured by other companies. This will be supplemented by the release of two new products later in the year.

The announcement coincides with the opening of the company's new factory in Hamburg, which follows the example set by Vao and Igeon to avoid the greater tariffs imposed by the European Community.

Traveling Secretary (24)  
 Ltd. Unit 14 Party 14  
 No. 14  
 Director  
 Chairman  
 Secretary

**Table 1**

A videotape that signals a new frontier in development therapy is part of an ongoing search for new talent. The development officer, Nina Frenkel-Prescher, is the latest recruit for David Greenfield who now coaches talented telegraphers in Boston. The deal promises a steady flow of games over 1997-2001. Two early assignments are late summer

The first of the new labels is a group of programmers who are all known to Armstrong through their personal links, with System 3 being the largest company of four. *Alpha* and *Beta* Hardware programmers John Twiddy, graphics artist Hugh Haley, and *Max/Disc* where artists include the composer of *Fast Alpha 2* is 250 frames. The new label gives the team a more positive profile and will hopefully bring them the recognition they deserve.

Malware Former Hosts in the second label and will support individual authors and development teams. The label will also be used by the development team, which will be appointed to produce games for the Nintendo Game Boy Advance.

Red Crosses Vice President of Astronaut Europe comments: "David has pulled together a very talented team and as such formed its strong an exceptional product line-up from New Frontier."

The addition of Wood Image Ltd and Winston Funnell House to our group is further evidence of the great strides which we have made over the last year. We continue to attract and retain a new talent which in turn strengthens our position within the

Timothy Anderson, Clerk,  
1st Dist. House, House  
Furn. Bldg. Reading Bldg.  
Rm. 214 Tel. 947-1147  
41-1000



# Graphics Guide

*What are graphics all about, and what do you need to get the best from your computer?*

By Kerry Fowler

A stunning graphic display is one of the most difficult effects to achieve effectively. This is especially true of the C64, but even the original Beavis of the C128, C16 and Plus 4 don't solve the problem totally. To get the best from commercial graphics programs are essential, and then the devices in which input device to use: keyboard, joystick, mouse, lightpen or graphics pad.

One of the strongest features of Commodore computers is their graphics capabilities. Sprites, characters and high resolution (if not combined with a fairly wide range of colours) can add impact to games and utilities.

Hi-res graphics are generally supported by most of the available packages, but sprite designers and character editors are few and far between. This is true so much so that a while a well-known programmer, Tony Crowther, published his own sprite and character defining routines in *Your Commodore* (Vol 1 Edition, August 84), the magazine had many



requests from professional software houses who wished to use the program to design their own screens. This led to an updated version being created for our sister magazine, *Commodore Disk* (Nov/Dec 1 Plus Nov, Dec, 85).

## User-defined Characters

Character graphics is often the first definition of the standard character set of the Commodore. The standard graphics are the letters and characters featured on the keyboard and they are all based upon an 8x8 grid of pixels. A pixel is the smallest unit of a computer screen display: it's a single point of light which can be variously coloured to give the effect desired.

Each pixel on a grid can be switched on or off. For example, to produce the letter 'A', a character grid would take the formation shown in Figure 1. The problem is that a Commodore video chip can only access 16K blocks of memory, as shown in the simple C64 block in Table 1. This means that the screen positioned at location 16384, can only

grab 4, character rows (locations 8 through to 16393). The computer circumstances that limitation by hardware memory, but the rule is one which can never be broken by the programmer.

With the screen at 16384, the problem is that a Basic program uses the memory on either side of the screen — below the screen is the workspace for the BASIC routine's variables, and above it is Basic RAM for the program itself. This leaves very little space for any user-defined graphics (UDG), so where sprites?

There are two solutions to the problem, the start of Basic memory can be moved up, or the screen itself can be placed in one of the other 16K blocks of memory. The mathematics guide to these methods is the relevant Programmer's Reference Guide from Commodore.

## Mighty Sprites

Sprites are like small independent screens which can only display 256, standard characters (maximum 8x12 pixels). If a sprite pixel is not turned on, the pixel becomes transparent and any characters behind the sprite will show through, but if the pixel is turned on, the sprite masks the background colour.

For some reason Commodore decided not to implement sprites on the C16 Plus 4, which probably helped to add to the machines' 'dreadful' because many of the programs written for the C64 involved sprites and couldn't easily be transferred to the newer computers.

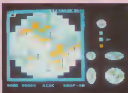
Characters and sprites can be displayed on a single coloured sprite or multi-coloured (single-coloured or standard mode) means that all of the turned on pixels are displayed in the same selected foreground colour, though this can vary from one character to the next. Multicolour is a slight nuisance, but it does allow three colours to be used.



TABLE 1

Table 1: C64 blocks for the VIC

Decimal	Hex	Block Number
49152-49155	C700-FFFD	1
32768-49151	8000-8FFF	2
16384-17267	4000-7FFF	3
8-16383	0000-3FFF	4



As with standard mode, the background colour can be different for each individual character or sprite, but in the same throughout that particular character. The two extra colours are the same for all of the characters on the screen. Although sprites follow the same rule, their extra colours are taken from different memory locations, which means that the two auxiliary colours may be the same for all of the sprites, but can differ from the background character.

The only problem with multicolour is that character definition fails to fill pixels and (2x2) pixels for sprites. This means that the fine detail is not as good as for standard mode, and multicolour characters always have a jagged-edged appearance.

## Higher Resolution?

High resolution is a misleading name, because the total number of video pixels is the same as in standard low-resolution mode. The difference lies in the freedom which is now given the user.

Hi-res graphics work on a similar principle to character graphics, and this is an important point to bear in mind when designing a screen display in this mode. Up to 256 UDCs can be defined in low resolution. At best, these would cover about a quarter of the screen if they were laid out to end, row by row. Hi-res allows a screenful of 1000 different characters to be defined.

The equivalent division among UDCs, would mean reducing the screen to a 16x16 indivisible grid of characters, because each of the 256 characters can only be defined once. This gives a pixel resolution of 256x256 instead of the full screen resolution of 640x200 pixels. Hence the justification of calling it high-resolution.

Each character on a hi-res screen has a fixed position, but the rules of UDC defining apply. This is especially notable in multicolour mode, because each full-character area can only hold a maximum of three colours. The extra colour freedom of hi-res does mean that neighbouring

characters can have three different multicolours, but the full pixel grid governs what can be done.

## Pack Selection

There have been many graphics packs for designing on the screen, but I don't know of any currently available packages for sprites or character definition which offer good value for money.

Trilogic's Expert cartridge system does have a sprite designer included in the package, but if this is the only facility which is of interest, then the price of £29.95 is a rather high one to pay. The best source of money is to check Four Commodore and to look for the numerous sprite and character defining programs which crop up from time to time.

Hi-res packages are shown in the Toolbox at the end of the article and typical features would include hi-res drawing, tiled and out-of-the-shape (concentric circles and rectangles), single and connected lines, text mode spray fill, solid fill, a range



of brush marks, and means for slowing it to add fine detail.

There are additional desirable features such as a window. This is the ability to select an area of the screen and repeat it elsewhere. This area can be used to redraw similar characters, because a character is "painted," moved and then modified using the repeat facility. To get the best from the graphics screen, a pattern fill routine of some kind can create convincing shading effects to add an extra dimension to otherwise 2D images. Patterning can also take the frustration out of creating repetitive shapes such as the bricks in a wall in wallpaper.

A final feature is usually called UNDO. This is most desirable when a FILL command goes wrong. Without UNDO, the image could be ruined beyond repair, with UNDO a press of a button can restore the image to its original state.

To fully utilize an image created by a graphics package the user needs to know where the various elements are stored. Since this is rarely in the normal screen position (HID) it is also necessary to understand screen banking, so a good graphics book is essential. For instance, the Programmer's Reference Guide is not very useful, but there are many guides available as good bookshelves, just check the index for a reference to banking screen moving or something similar.

## Graphic Aids

For many people, the cheapest tool for graphics is the joystick because it can be used for graphics or gameplaying. My own preference is a mouse, because a mouse can be more easily operated with one hand. Even if this only means that the other hand is free to hold a cup of coffee. I find the use of a mouse most relaxing.

Keyboard control is another consideration, and sometimes offers the most accurate control system. Unless you are a touch typist, the problem is that it is tedious to watch the keyboard with one eye, and the screen with the other.

Several packages offer lightpens as input devices. In my experience, these are wonderful devices which most closely relate to drawing on paper, but almost lack the accuracy of joysticks and mice. The problem lies in the manner in which these devices work.

When a television or monitor drives a screen, the image is created by a stream of electrons being scanned

across the lines of a screen. When an electron hits the phosphor coated screen energy is dissipated in the form of light. The greater the number of electrons, the brighter the light given off.

The electron gun therefore varies the number of electrons aimed at any point on the screen to create the light and dark areas of the image. In colour monitors there are at least three guns which represent the primary colours which can be mixed to create the full range of hue and tone which makes up a colour image.

Each gun starts at the top left of the screen, and scans across the top line. Then it flies back to the left end of the next line and scans that next row. This continues until the bottom line is reached and completed, then the gun is at its retracted back to the top left of the screen, and the process starts again.

Each point on the phosphor screen acts like a red-hot poker which has just been removed from the fire - it glows. The fade actually only takes a very tiny fraction of a second, and the first dot may be extinguished before the next reaches the bottom line, but the whole process is so rapid that the eye cannot perceive this.

A computer calculates the screen scan and can calculate where the gun is pointing at any particular time. If a light screen is placed in front of the screen, it can calculate when the phosphor area in front of it is caused by reflecting the light that burns forth. The calculator can then calculate the time delay, between triggering the screen scan, and the light slowly detecting the effect of the gun passing a particular spot. This allows a grid in the computer's memory to be written and operated on, according to the distance of the software running at the time.

The problem is in getting pixel accuracy. Some software is severely affected by interrupts which can delay writing of the lightpen sufficiently to displace the perceived position of the pen, once the calculation has been done. This normally manifests itself as a twinkling of the cursor on the display screen or a line becoming broken or broken as the pen is moved across the screen.

Graphics pads work on a different system. Imagine a crosshairs grid of wires which are embedded in a plastic medium. The wire and web of the grid is separated by a fraction of a

millimetre, so that slight pressure will depress the two wires. The net effect is one of a flattened rail, vertices which correspond to each pixel on the screen. This is analogous to the solution made a graphics pad.

Once the signal is generated, internal electronics calculate which switch is operating, and send the information to the computer, which then acts accordingly. The problem with a pad is that sometimes the point of the stylus may rub against between two switches, and both are triggered. Usually the internal electronics will make an excessive distance, but sometimes one of the switches may constantly make and break and the screen cursor dances back and forth in sympathy.

On the subject of pads, most software is produced for the Koolaid Pad, which is well available in the States, but try getting one over here! It's impossible Macintosh market. Sometimes misleadingly named, Amstrucore Joystick, which looks similar to the Koolaid but is totally incompatible with Koolaid software. Based on experiments with Baudard's GCP Studio. Fortunately, it does come with its own software in patchy version of Missing Pixelsoft, which is fairly comprehensive and the package is highly recommended for those who would like to try a graphics pad.

Whatever device is used, graphics are the root of all successful programs. Text is fine and neat, in contrast but an illustrated manuscript is far more impressive than a paperback!

## Peripherals

**Flashing Paddles £12.99** Data Electronics with lightpen £24.99

**Graphics Support Utilities Disk £12.99** Data Electronics

**The Advanced GCP Art Studio £24.99** Baudard Software

**4x16x4 £29.99** Baudard House

**GEOS (Greenland) Disk only £24.95** Graphics Master (see with Amstrucore disk drive)

**NEOS Mouse £24.95** Data Electronics

**Campana**

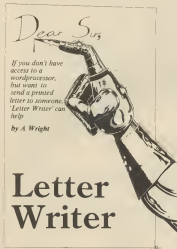
**Disk Mouse £24.99** Data Electronics

**4x16x4x4 Station 4** (Amstrucore)

**Macintosh**

You may have noticed that there are all for the C64 GEOS incorporating GEOSport is available for the C128 at £19.95 but there appears to be nothing for the C16, Plus!





*If you don't have access to a wordprocessor, but want to send a printed letter to someone, 'Letter Writer' can help*

*by A. Wright*

# Letter Writer

**H**ave you ever wanted to send someone a short letter that looked official? I'm sure many of us have sent one of those groveling-petty-please-type letters at some time or another. Well, the letter will look less of a repetition after all, if your handwriting looks as though it's been printed by a spouter that spouts the previous writing in a slightly better, you're not going to get very far, are you?

This is where 'Letter Writer' comes to the rescue. It's not designed to be a wordprocessor program; it simply allows you to compose your letter quickly and then print out the finished masterpiece. One last note: The main menu of 40-column wordprocessor programs

will appreciate the fact that the display is in 80 columns and then you can see each line on screen as it would appear on paper.

## Getting It All In

**F**irstly, it's important to note that you must have a copy of the "ROM 80" program by Jim Meyer, which puts the C64 on 80-column screen display. That appeared in the January 1988 issue of *Four Commodore*. Also note that both programs can be entered with or without the "ROM 80" program resident in memory.

Type in Letter Writer and SAVE it to tape or disk, then type in Letter

Printer, and SAVE it immediately after I find it best to have the programs saved immediately after each other in the following order: ROM 80, Letter Writer, Letter Printer.

## Using the program

**I**f you haven't already done so, LOAD and RUN the ROM 80 program. Next, LOAD and RUN Letter Writer. If all is well, you'll see the following: at the top of the screen is a copyright message; below this, on the left, is a row and column count. Taking up the centre of the screen is a large box, this is the "window" through which you will see your text. At the bottom, is the message "PRESS 'CTRL' FOR HELP SCREEN".

Type something (apart from 'CTRL')—you can enter anything except Commodore keyboard graphics, and the left and up-arrow keys. Notice that if you enter quotes, they don't appear, but an up-arrow does this is because printing quotes can have a word effect. Don't worry! The arrows are exchanged for quotes just before printing.

Now, when you enter text, the program has to check a list of things before it PRINTs each character, and these checks slow the program down a little. As a result, if you enter a fast typed, you may notice that the screen display doesn't quite keep up with you. This shouldn't be a problem, as the characters are stored in the keyboard buffer. However, the buffer only holds ten characters, so if you type fast, some characters may be missed. The only solution, I'm afraid, is to slow down!

Right, on with the show... pressing RETURN, or reaching the end of a line, will move the cursor to the beginning of the next line. Note that if there is any text on the next line, the cursor will move to the end of that text. Pressing DELETE will erase the character on the left of the cursor. If you are in the first column of a row, pressing DELETE will move the cursor to the line above, among the last character on that line. If you wish to move to the line above without deleting the last character, press the up-arrow key. The cursor will move to the correct position on the above line. If you move up or down on to a line which already contains 40 characters, the cursor will be in column 41, which is not normally possible. In

this situation, you cannot enter a character. You can only move up or down.

Now, when you've reached the bottom of the line, and you **RE** the line as given, **RETURN**, the text will scroll up by one line. When you are on the top line, and press the up-arrow key or delete past the beginning of the line, the text will scroll down by one line. Note that you may only have up to 60 lines, and you can't scroll up past line 1 or down past line 60. I chose 60 lines so that it is as easy as can fit neatly onto one sheet of A4 paper.

Using BASIC, the scroll commands are quite slow. If you find it's too slow for you, I suggest the following:

When you reach the bottom line, and have entered the text for that line, press **RETURN** a few times (10 or more). Then, when the scrolling has finished, move the cursor up to the correct position and continue typing.

## THE OPTIONS

Now that you know how to enter your text, it's time to learn what you can do with it. The first thing to do is press **CTRL** to access the help screen. You must go to the help screen before you can use any of the options.

The top of the screen contains a simple reminder about how to enter your text; the options are at the bottom. To select, simply enter the number which is to the left of the option you require. The options are:

### 1 RETURN TO TEXT

The option does not affect the text in any way. It simply returns you to where you were before calling the help screen. It is the only option which doesn't ask "Are you sure?" before continuing.

### 2 GO TO LINE

What line to go to, what line does? After making sure you're sure, simply enter the number (1-60) of the line you wish to go to. You will be returned to your text, with the cursor at the correct position on the line you selected.

### 3 SET TEXT NAME

This option allows you to enter a filename which is used when saving or loading your text. When you first RUN the program, as if you enter a

blank line, the current name is given as "No name!" i.e. the program will save or load for an as-named file, not one with the name "No name!" To give a filename, type in the name and press **RETURN**. The same ways the name unless you enter another, or select **"DELETE SHEET"** (see option 5).

### 4 MOVE TEXT

On selecting this option, you are asked for the start and end lines of the block you wish to move, and the new start line of the block. When you've entered these, the block will be moved, and you will be returned to your text at the line you were at before calling the help screen. You should be aware of the following things when using this option:

- You **CAN** move a block to start within itself, i.e. if you move text from lines 1-20 you can define the new start line as line 10.
- The block to be moved must fit, i.e. you can't move a block of 20 lines to start at line 50. If you try this, the block will not be moved, and you will get the message "Block will not fit" and will be returned to your text where you left it.
- To move only one line, enter the same number for both the start and end lines.
- Most importantly, moving a block will over-write any text that is in the new position. Also, the lines are deleted from their old position.

### 5 COPY TEXT

This works in exactly the same way as "MOVE TEXT", except that the lines are not deleted from their old position.

### 6 DELETE TEXT

On selecting this option, you will be asked whether you'd like to clear a block or the entire sheet. Selecting "block" will prompt for the start and end lines of the block to be cleared. The block will then be cleared, and you will be returned to your text at the correct position. If you select "sheet", the program is **RE-RUN**. This clears all text and resets the filename to "No name!"

### 7 SAVE TEXT

It's now there's no need to explain this option, but I will tell you that the current filename is displayed along with the "Are you sure?" prompt.

Therefore, if you wish to change the name before saving, just press "N" and go to the "SET TEXT NAME" option. When the text has saved, you'll be put back where you were.

### 8 LOAD TEXT

Again, there's no need to explain this. Note, however, that when text has loaded, the cursor will be on line 1 of the text. If you can't remember the name of the text you wish to load, go to the "SET TEXT NAME" option and enter a null string (press **RETURN**). The program will then load the first file it comes to. Also note that when a file is loaded in this way, the name is **NOT** set to that of the file, i.e. it is left blank (or "No name!")

### 9 QUIT PROG

This is self-explanatory.

## PRINTING OUT YOUR TEXT

I had to write the printing program as a separate unit, as my printer is non-Commodes. And I have to use an interface badly, my interface software uses the same area of memory as BASIC 10. (For those who are concerned, my printer is an Epson MX-80 F.T. Type III, and my interface is the "Commodes Connector", which I bought in Bonn for £19.95 about two years ago. It does its thing, but it suits my needs perfectly.)

My printer defaults to the English character line on power up, and while in this font, printing a backslash (\) will result in a pound sign being printed. So if I want to print a pound sign, I send a backslash to the printer, hence line 8 of the Letter Printer 2 program. If I want to print a backslash, I must change to the American character font, send a backslash and change back to the English font, hence line 11.

Costly? You should be! Let me make an example of your printer can print a pound sign, AND a backslash from the SAME font, define lines 9 and 11 from the Letter Printer 2 program. By the way, a list of the printer control codes used can be found at the end of this article.

Right. When you've used your text using the Letter Writer 2 program, switch off your computer, connect your printer, and watch them both go. Load your interface software (if necessary) and then **LOAD** and **RUN** the Letter Printer 2 program. Follow



# THUNDER BLADE

One of the hottest out-of-pocket games of the year, and the sequel to Sega's *Afterburner*, *Thunder Blade* is now competing head-to-head with that high-flier for the so-called Christmas number-one spot.

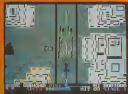
As you push you behind the wheel of the Thunder Blade, an advanced attack helicopter that's armed, geared, and rounded and armed with a 1,500-round weapon system, you're in a streamlined machine. Unfortunately, in the C64 version it's hard to find any difference between these two examples, and they seem to appear randomly on the screen, changing anything and everything.

Your mission is to destroy the advancing forces of an antagonist who is threatening your land. This gives you comparatively no time everything in sight as you battle your way through the game's four levels. Each level consists of a north-top-down and 3D view. Combat sequences in which you shoot at and with enemy attack helicopters, Bi-geat Tiger jets, Baranade torpedo-cruisers, Carrier Sphair aircraft,

the tanks, the view switches to a 3D perspective, and you have to run the gauntlet as Skyraider very soon again. This time the enemy helicopters are more effective, and you get your first chance to swap some *Afterburner* style controls.

If you survive that you win the chance to battle the first super-forest, a giant warship bristling with guns and missile launchers that will test your skills and Thunder Blade's manoeuvrability as you hover back, and forth, dodging the defenses. A bar at the bottom of the screen displays important data such as your score, hits, the high score, your speed, and how far you've managed to go through the stage.

Stage two takes you out of the city, and into the mountains and deserts. The battle remains the same, but now some vehicles replace the helicopters. At the end of this level, a massive troop carrier forms the super-forest, and presents even greater challenges.



armoured cars, and Baranade aircraft. It's kind of obvious, though, that it's only blowing them out of the clouds. The end of each level is greeted by an empty, armed super-forest, with a single helicopter to chase a sequence of Thunder Blade.

In first place is in the graphics, people of Skyraider City, and before with a top-down view as you take out from the Forest it's a help. You're also being dodging fireballs, instead of your first tanks, armoured cars, and even the building themselves. The graphics at this point are very disappointing: the buildings are just four-colored squares that are a mess, over each other to give you the bad impression of a full-screen block. While I appreciate that programmers need to take shortcuts to keep the speed at the game to go, it's level this just set up to the high standards, so have come to expect.

Having said that, the gameplay will drive you back for more and more. Once you've cleared the big car and released

Stage three and four take you down the river drive to take the Baranade torpedo-cruisers, and then the refinery, to deal with a Bi-geat carrier, and the ocean and super-forest, the ultimate test for advanced pilots.

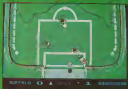
*Thunder Blade* was a top arcade game, and a almost guaranteed great success as a coin-op conversion. But of all the available versions, the C64 one seems to be the worst. Although the game play is good, the graphics are poor, and the sound and music have the same effect, and you get only three Thunder Blade's game, whereas the other versions give five. It's good, but it could have been so much better.

## Thunder Blade

Title: *Thunder Blade* Supplier: J & Gold (over 2) (Hoford) Wai Waiford, Devonport St 74Y Tel: 021 338 333 Price: £9.99







# A Flow of Ideas

*Formal flowcharts are the best way to show how a program works*

*By Norman Doyle*

As I look at programming aid tell you that BASIC statements can be used to remind the programmer how a program was originally conceived. Having gone back to modify many of my older programs, I have found that BASIC only works to a certain degree. The best answer is to create a formal flowchart, because this can aid conversion from one machine to another as well as later updating.

A flowchart is a graphic display of the logic behind a program and is the old adage given a picture is worth a thousand words. Formal flowcharts or flow diagrams as they are also

known, rely on a series of flow symbols which represent certain specific actions within the program. All flowcharts should start and end with a terminal



To indicate the start, the terminal has START written inside it, and the end terminal has STOP as it was long-winded.

Some flowcharts spread across several sheets of paper, so a special *off-page connector* is used which contains the number of the page to which it connects. Similarly the connector on the page to which it refers starts with an *on-page connector* containing the number of the page from which it continues.

13

Probably the most used symbol in a program is the input/output parallelogram (I/O symbol). This contains such things as print statements, input requests, DATA reads, print out statements and disk or tape access. The Data manipulations

are contained in rectangular processing symbols. The contents include the handling of strings and logic operations, as well as the display of most conventional mathematical formulae.



Sometimes a program can branch as the result of a decision. A branch is denoted by a diamond-shaped



Notes the arrows on the branch lines. These are flow-direction indicators and should appear between every line to indicate the direction the program is taking because the direction of flow may not always be obvious.

Sometimes would be difficult to show on a single diagram so a special form of the processing rectangle has been devised to cope with this. The *perforated process symbol* looks like this:



In which contains all of the program information, but there may be systems where entire information is necessary. That applies to the REM statement in Basic. The annotation symbol is an arrow-shaped box which connects to the flowchart by a dotted line to show how to which it refers.

```

1000 1000 = 1000 - 1
1001 1001 = 1001 - 1
1002 1002 = 1002 - 1

```

A typical flowchart looks like the one in Diagram 1. Notice that the branch represents the drawing of a loop line which connects the decision symbol back to the re-entry point. In a program with several of these links, the page would soon look very messy, so a connector symbol, box lines drawn which is only used to connect two parts of a flowchart contained on a single page. If a branch goes over to a new page then the all-page connector is used.

This is the stage of symbols commonly used in flowcharting. Professional diagrams also use another set of symbols for various storage devices and communications links, but these would rarely be needed in a small home computing environment.

## Exploded Diagram

Diagram 1 shows a box to decimal conversion. It's obvious where the program starts, and the first operations is to zero any necessary variables and then print a request for a number input onto the screen. Although Commodore Basic allow INPUT statements to print the text, and the input prompts on the same line, this may not be true in all cases, so the output of a number string is based as a separate action.

The input is then tested to see if it starts with a dollar sign. If it does the program continues otherwise a return for a number input. At this point the program could include a decimal to hex converter as a future development. Although the full branch line leads to the start is shown, in this case it could be replaced by the use of connector symbols as used on the rest of the diagram.

Following through the conversion, the program next checks if the string is five characters long (number plus significant characters). If this is not the case then the program loops back to position 3. If true, the conversion process is started with a loop which takes each character in turn, multiplying the dollar sign by using a MULTI command.

Each character is tested for validity and if any one fails then the program loops back for a new input. Valid characters are passed to a subroutine called number conversion, which has an annotation symbol attached to explain what the subroutine does.

The subroutine actually converts the hex character to its decimal equivalent and returns it as the variable VALUE. This variable is then added to variable DECIMAL and then multiplied by 16 to prepare it for the next hex place.

The loop is checked to see if all values have been taken and if not, the next value of 4 is taken. If the loop has ended, the program prints the message DECIMAL = and then the decimal value. Note the usage of expanding text and variable values. Text is enclosed in single quotes and variables as a plus name.

The program then by asking if another input is required and loops back if that is the case. Otherwise, the program ends at the stop symbol. Converting the hex program is simple. Try it for yourself and then check the listings page to see how I did it.

## PROGRAM LISTING: CONVERSION

```

10 10 0-0
20 20 INPUT ENTER NUMBER IN HEX
30 30
40 40 IF LEFT$=0 THEN GOTO 1000
50 50 IF LEFT$=1 THEN GOTO 1000
60 60 IF LEFT$=2 THEN GOTO 1000
70 70 IF LEFT$=3 THEN GOTO 1000
80 80 IF LEFT$=4 THEN GOTO 1000
90 90 IF LEFT$=5 THEN GOTO 1000
100 100 IF LEFT$=6 THEN GOTO 1000
110 110 IF LEFT$=7 THEN GOTO 1000
120 120 IF LEFT$=8 THEN GOTO 1000
130 130 IF LEFT$=9 THEN GOTO 1000
140 140 IF LEFT$=A THEN GOTO 1000
150 150 IF LEFT$=B THEN GOTO 1000
160 160 IF LEFT$=C THEN GOTO 1000
170 170 IF LEFT$=D THEN GOTO 1000
180 180 IF LEFT$=E THEN GOTO 1000
190 190 IF LEFT$=F THEN GOTO 1000
200 200 RETURN

```

# The Software Spellbook

*In which Wizard Emeritus Myron Patch, Grandmaster of the Honourable Order of Coders and Algorithmicians, offers essential lore for the apprentice Numeromancers*

There is a CMA without a few easy and spells (the operations refer to these as "software") is an operation as trying to compile the binary code, without spelling the field of a tape run. Therefore, in this sense, I offer to those who have the run that can take as much wisdom, a few small fragments of forbidden lore that may dramatically increase your ability in the arcane branch of knowledge known as machine coding.

I assume that everyone out there knows the Fundamental Code Level spells such as Addition and Subtraction. Without these, of course, no progress on the more arcane level of Multiplication and Division is possible. It is the first of these that we shall be looking at.

The second form of Multiplication spell is understood as shown in binary (with a flourish). This multiplies two single-byte numbers (NUM1 and NUM2) to give a two-byte result - RESULT. Already, this shows us something unusual about Multiplication. You must always, when multiplying a number of *n* bits by another of *m* bits, allow for a result of *n+m* bits long. This is a general principle of software magic, which the arithmeticians at this point

do not know the spell work? All apprentice grade spellworkers already have a grounding in the popular but vulgar magical language known as decimal, reckoning being the number 10. You may be surprised to find that performing multiplication spells in this language is actually more complicated than using a method based on the magical law of 2, which we casually call binary.

Here is a typical box you multiply

	20	0001
	040	
3x35	185	
CARRY	1	11001
	115	
4x23	82	
CARRY	1	
	002	
CARRY	1	
	100	

The more essential components of this are used in binary. You can see that there is a process of repeated addition, but that, because of the arcane system, each addition takes place on a level 16 times larger than the previous. This is achieved by shifting each succeeding sum one position to the left. Secondly, note the use of the carry. This, of course, is used when a multiple goes beyond the range of one column (i.e. greater than 9).

All that is obvious, but my reason for explaining it should become clear when we repeat the same process for two small binary numbers.

	111
	1001
111x1	111
111x0	000
111x1	111
	—

Now, could this be simpler than that? For a start, each multiplication of the first number (the multiplicand) can only give rise to either the number itself or zero. There are therefore no carries in the multiplications, since the range of one column is 0-1 and you cannot get two no matter how you try. The carry comes in in the addition, but once again, a carry can only be a 1 or it doesn't exist at all.

Of course, we do write rather differently on paper from the way that the hardware will do them. While we could design a machine code routine that would do that trick exactly as we just did it, that would not be efficient. Most algorithms are long structures, so a machine code routine does not join up a long of multiply words and add them up at the end. It instead performs an addition to a cumulative total every time it goes over around the loop.

The process is like that for 011x2

1. Is leftmost digit of 011 a 1? Yes, it is, done by shifting 011 leftwards, and it's a 1, then the carry flag will be set. If it is not then proceed to 3.  
2. It's a 1, so we have our multipliers. That of course just happens to consist of adding 011 to a result field (this should have been cleared to zero at the start of course). The addition may





# Pac- mania



Unless you happen to be a former living in a cave in the unpopular end of Great Mongolia, you will know and either love or hate Pac-Man. That yellow triangular character that while he maintains a serious mien at an alarming rate has always starred in two computer games, the original desktop adventure and the game of the decade sequel *Pac-Mania*, which was a sideways-scrolling arcade adventure featuring ghosts in places and other equally unlikely events. Now, Pac-Man's back in 3D.

Pac-Mania is not new, much like was through 3D scrolling arenas filled with dots to eat, ghosts to avoid, and power pills to turn the ghosts pale and into bonus points as they scuttle back to their lair to recharge. As in the original game, Pac appears for a limited time to sweep you into the middle for a juicy bonus, but also into the clings of patrolling ghosts and away from the power pills.

Pac-Mania is more than just a 3D version of the original game, although that would probably be enough for most Pac-maniacs as it adds a few extra features that will make the game a challenge for all players, whatever their skills. From the opening screen you can select which level you will start at, and either start slowly with the easier scenes or jump straight into the thick of the action.

The game consists of four different screens that you visit again and again as the things speed up and generally get tougher: that go under the misleading names of Black Trees, Pacman's Park, Scythe Land and Jumpy Steps.

Chide the chad ghosts, has retained two new spikes, two and finally, to protect you, but Pacman has a new trick up his sleeve - pressing the fire button makes him jump over ghosts of enemies to escape from tight corners. Pacman also has the help of two new power pills that appear in place of some fruit. A red power pill gives you double points, which can really help to rack up those high scores, and a green pill that temporarily makes charges Pacman without he bumbles around the maze.

Pacman has always enjoyed a cult status, and has been played by those who don't normally play computer games (such as Your Computer's production people) and with the addition of 3D graphics the following should give and work.

## Teacher

**Title:** Pac-Mania. **Supplier:** Grandstream Entertainment, 12-18 Paul Street, London EC2A 4PS. **Price:** 39.95.



Boxer 1	Weight	Stance	Hand	Height	Reach
Boxer 1	168 lbs	Orthodox	Left	5'10"	74"
Boxer 2	168 lbs	Orthodox	Left	5'10"	74"
Boxer 3	168 lbs	Orthodox	Left	5'10"	74"
Boxer 4	168 lbs	Orthodox	Left	5'10"	74"
Boxer 5	168 lbs	Orthodox	Left	5'10"	74"
Boxer 6	168 lbs	Orthodox	Left	5'10"	74"
Boxer 7	168 lbs	Orthodox	Left	5'10"	74"
Boxer 8	168 lbs	Orthodox	Left	5'10"	74"
Boxer 9	168 lbs	Orthodox	Left	5'10"	74"
Boxer 10	168 lbs	Orthodox	Left	5'10"	74"

**Boxer 1**

**Boxer 2**

**Boxer 3**

**Boxer 4**

**Boxer 5**

**Boxer 6**

**Boxer 7**

**Boxer 8**

**Boxer 9**

**Boxer 10**

# T.K.O.

**Boxer 1**

**Boxer 2**

**Boxer 3**

**Boxer 4**

**Boxer 5**

**Boxer 6**

**Boxer 7**

**Boxer 8**

**Boxer 9**

**Boxer 10**

T

Boxing is a sport that has been around for centuries. It is a sport that is both physically and mentally demanding. Boxers must be in excellent physical condition and have a strong mental game. The sport of boxing is a combination of strength, speed, and technique. Boxers must be able to take a punch and deliver one. The sport of boxing is a test of endurance and willpower. Boxers must be able to fight for long periods of time without tiring. The sport of boxing is a challenge that only the best can overcome.

the 1990s, the industry has been hit hard by a combination of factors.

First, the industry has been hit by a combination of factors. The first is the fact that the industry has been hit by a combination of factors. The second is the fact that the industry has been hit by a combination of factors.

The third is the fact that the industry has been hit by a combination of factors.

The fourth is the fact that the industry has been hit by a combination of factors.

The fifth is the fact that the industry has been hit by a combination of factors.

The sixth is the fact that the industry has been hit by a combination of factors.

The seventh is the fact that the industry has been hit by a combination of factors.

The eighth is the fact that the industry has been hit by a combination of factors.

The ninth is the fact that the industry has been hit by a combination of factors.

The tenth is the fact that the industry has been hit by a combination of factors.

The eleventh is the fact that the industry has been hit by a combination of factors.

The twelfth is the fact that the industry has been hit by a combination of factors.

The thirteenth is the fact that the industry has been hit by a combination of factors.

The fourteenth is the fact that the industry has been hit by a combination of factors.

The fifteenth is the fact that the industry has been hit by a combination of factors.

The sixteenth is the fact that the industry has been hit by a combination of factors.

The seventeenth is the fact that the industry has been hit by a combination of factors.

The eighteenth is the fact that the industry has been hit by a combination of factors.

The nineteenth is the fact that the industry has been hit by a combination of factors.

The twentieth is the fact that the industry has been hit by a combination of factors.

The twenty-first is the fact that the industry has been hit by a combination of factors.

The twenty-second is the fact that the industry has been hit by a combination of factors.

The twenty-third is the fact that the industry has been hit by a combination of factors.

The twenty-fourth is the fact that the industry has been hit by a combination of factors.



# Risk



# Operation Wolf

There has been a dearth of good shooters up this year, but now your trigger finger can stop itching. *Operation Wolf* has arrived! Fix and away the most successful arcade game of recent times, the only thing that Demco have not managed to include in their conversion is the 100-balls machine gun that sits in front of the arcade console.

Basically, you're on a one-man mission of death and destruction. First, you have to destroy some prisoners towards the end of the game, but think of all the bodies you can pump full of lead as you go. This is definitely a game for someone with the mentality of a Rambo rather than a General Patton.

The game comes in an horizontally scrolling stage. In each section there is a distribution of forces that has to be taken out before you can progress. These forces are not only of enemy soldiers, but also tanks, armoured helicopters, and it is for this reason that you are equipped with a handful of rockets as well as clips of ammo for your gun.

Considering the nature of your mission, you go into battle decidedly disarmed, with only one clip of ammo and five rockets. Extra weapons do appear on the screen which you can pick up if you shoot them, but there is frequently a conflict of interests here—do you take out the heavy enemy forces and risk running out of ammo, or risk life and limb instead?

Because of the paucity of ammunition, it's not clever to keep your finger permanently on the fire button. Instead, retreat yourself to short bursts. Longer bursts are required to dispose of the tanks and helicopters should you run out of rockets, but this is worthwhile as they do most damage to you. Other objects to look out for as you blast your way through stand the screens are open boxes of dynamite.

Shooting these will effectively destroy everything on screen, so take your time before blowing it up. You can also gain automatic fire power, but this lasts for a short time only.

As you progress through the levels, so the difficulty of your task increases—not only do you have more men to kill, but the accuracy of your aim needs to be spot on as well. The large soldiers start wearing bullet-proof vests, and have to be shot through the head rather than the body.

Naturally, you're not indestructible, and an enemy level shows your current state of health. You can improve your state of health by completing a level or shooting a power pill but still, the ticking scale moves necessarily towards your impending death. The best way to postpone your impending doom is by making sure that you take out the vehicles and paraphernalia as quickly as possible. Also, shooting innocent civilians such as the nurses is considered to be a definite no-no, and you are severely punished for such misdeeds.

Instead of the machine gun at the arcade version, using a submachine gun conserving a precious clip round the screen. Jayhawk response is excellent, once you've got the hang of what's going on. There is also the option to use a Nixon name which is smart to use, although to compensate you get less ammunition to start with.

The conversion is on sight, but machine is excellent, and all the addictive qualities of the arcade machine are present. Slightly verbose it might be, but it is great fun as well. Strongly recommended.

## FinalWord

**Title:** *Operation Wolf* **Supplier:** Games Price £9.95 (cart)  
£14.95 (disk)



# Extending Basic

*In this second article on Extended Basic, we develop a COLOR command and GOTO and GOSUB variable*

*By Burghard-Heery Lehmann*

**T**he main disadvantage<sup>1</sup> with Commodore BASIC is that it doesn't give the programmer any access to the better features of the CIB. There just isn't even a command that lets you change the screen colors.<sup>2</sup>

In the first article of this series on extending the BASIC of the CIB, we introduced a simple command that changes the border colour. We can now expand on this and introduce a **COLOR** command that lets you change the ink, the paper and the border colour, all with one command

## 'COLOR'

For the command itself I've chosen the American spelling, which is 'color', because the screen and the letter is type in and save as five bytes in the machine code routine. Remember the last two letters of the command will be interpreted by the computer because CIB is a Basic function which has the letters for 88 (decimal 785) because of the 1 compare in line 880 of four program with 880 instead of 0 and 8.

The syntax of the **COLOR** command is **COLOR #1, #2, #3** whereby #1 is the value for the ink colour, #2 is the value for the paper colour and

#3 is the value for the border colour. As you can see, the separator between each parameter is a comma, even though I haven't included an error check in the routine that forces the user to use a comma and only a comma. This might be a job you like to try your hand at. Here is a clue to force the routine to give a System Error report jump to 54108. This will load the X-register with 508 and then print the System Error message.

As I promised in the last article we're going to make the use of variables possible. In order to make this possible we are two ROM-routines: **IADRA**, which evaluates an expression, and **SHFT?**, which translates a value from the floating point accumulator #1 into the two page variables \$14-15. **IADRA** gets an expression from the Basic text and evaluates it. The value of the expression (as "variable") is then put into what is called the "floating point accumulator" (short "fp-acc").

## The Floating Point Accumulator

There are two floating point accumulators (p-accs #1 (\$14-15) and fp-acc #2 (\$58-59). These are used by the CIB for all the

mathematical calculations it does. For example, if an addition is to be done the first number is put into fp-acc #1 and the second number is put into fp-acc #2. Then the addition itself is done. Finally, the result is left in fp-acc #1.

Each floating point value is stored in five bytes, the first byte represents the exponent of the number, the second byte holds the sign (+ or -) and mantissa 1 and the final three bytes hold mantissa 32 to 4. This makes it possible to hold floating point numbers of any size: negative as well as positive, and do calculations with them with a high degree of accuracy. But it is also quite cumbersome and slow. That's why it's usually better to develop your own mathematical routine unless there are the subroutines in ROM.

Nevertheless, in our routine it is convenient to use the ROM-routine because not only does **IADRA** interpret a simple variable but it can also do a mathematical expression. The routine is in Basic, for example, **COLOR INC=1 PAPER=0 BORDER=2 SHFT?** gets a value from fp-acc #1 and puts it in register \$14, which all we need is one page \$14-15 and the Y register (see line)

and the *Journal of the American Medical Association* (JAMA) (1997).

All we have to do then is load the contents of the T-register into the system variables which set the colour, that is 540 for the ink colour, 5720 for the paper colour and 57260 for the border colour.

### GOTO with variable:

The next two numbers (lines 1000-1100 and 1170-1180) enable us to enter GOTO and GOSUB with a variable. That is, you can enter a line like "SUBROUTINE 1000" and then "GOSUB SUBROUTINE" (BASIC remembers all with all variables so Commodore BASIC only lists the first two lines of the variable range; see talking with BASIC).

To modify GOTO so that a variable or expression is accepted is really very simple: since the token for GOTO is line 39, we compare it line 40 with that value; and, if it matches, make the program flow jump to our source line 1000-1170.

First, we point to the first character of the parameters of the EOTO command by calling CHARGPT (see 1005). The next two lines (1006-1008) do the same thing as we did with our COLOR command: the variable in parentheses is evaluated (SANDA) and the value which results is put from file page # 1 into core page 514/5 (SANDP).

Finally, we call the `EDIT` routine at `SALA` which executes the `GOTO` command itself, that is, makes the flow of the basic program jump forward or backward to the line specified after `GOTO`. This routine expects the target line number to be in core page `E4`. (2) That's why gathered the value from file name #1 into these variables, with the `SWCH` routine earlier on.

**GO SITE** with us below

**GOBACK** (line 1170-1180) is a few more complicated. First, the relevant information for the line to which the program flow has to return - when the RETURN command is given - has to be stored in the machine stack, and then an auxiliary GOTO has to be performed.

Because of GUSAR, I had to modify the overall structure of our Extended Basic program slightly as well. You may have noticed that in the beginning of the Basic section, Dave ROOM systems are used first, CHARLOTTE is called, to get the index of the command itself as the Eval

character of our extended command (line 38). Then our main routine is called (and not jumped to, as in the program in the previous article—line 39), and finally a jump is made to SATAP (line 40), that is, the ROM routine which executes the next basic statement.

In other words, our main reaction (lines 440-450) has been made wholly into a subordinate I copied lines 1180-1190 of the 224728 edition from the ROM edition of 24.812. I couldn't even add a PR to the

## Future Work

[illegible]

**Abstract**

### The Machine Costs

[illegible]



# Basic Workshop

**B**asic Workshop is a program which lets you press two keys (CTRL + ?) to print a keyword such as LIST. It allows the Commodore keyboard to be used in much the same way as the function keyboard it is copied at PC800 CB80, so as not to get in the way of Basic programs.

It works by using mnemonics and the keyboard matrix. When the mnemonics are reversed, the program checks to see if a key is pressed. If so, it stores the string in the keyboard buffer. The strings are held in a table and most in a language programmer should be able to change them with ease. The programmer can also change screen colours with the function keys, and has an inbuilt 'OLD' and 'Reset' This allows you to examine programs such as 'Pacman' by Commodore by installing the program, scrolling and OLDing the program.

It was written on the Monitor Assembly on Action Replay IV. A full list of commands are, follows.

## LOADING

Type LOAD BASIC WORKSHOP:1

Then NEW

Then ST549902

If this is all done when entering lines all you will get are OUT OF MEMORY errors. It should now print a message showing that all is well.

## CONTROLS

All the controls can be used by pressing CTRL + ? to show a help sheet:

F1 - Change colour

F2 - Screen text colour

F3 - Screen background colour

F7 - Screen colour

Shiftlock - Listing, direction, etc.  
Pause

*Avoid typing all your keywords out in full with this ingenious new program*

*by B. Graham*



CTRL + A = ASC

CTRL + C = CLOSE

CTRL + E = SEEK

CTRL + H = CHR\$

CTRL + L = LIST

CTRL + P = POKE

CTRL + R = RETURN

CTRL + T = DATA

CTRL + B = GOSUB

CTRL + D =

DIRECTORY

CTRL + G = GOTO

CTRL + I = INPUT

CTRL + O = OPEN

CTRL + Q = LOAD

CTRL + S = SAVE

CTRL + V = VERIFY

CTRL + = = OLD

CTRL + [ = HELP

CTRL + . = NEW

CTRL + (LBM) = RESET

When OLDing you MUST press CTRL and + and not type OLD. This is because it is the keypress which

OLDS not the actual word. Thus you get a Syntax error if you type it in.

# Games Update

by Andrew Oakes, *Editor*

**B**est-selling computer games are a UK success story, and the genre is growing a collection. There are now many versions available for the PC, the latest of them is *Jeopardy!* game, being endorsed by Dick Wallace.

Nevertheless, different games are on offer. Four players share an number of players to compete in a fast paced game. "Two player head-to-head" allows just two players to team up against each other and "round the clock" is the old favourite where you have to be the numbers in sequence. If playing round the clock you can specify if you need to be just the number, either as doubles in order to pass onto the next number in the sequence.

Once you've selected your playing options and are ready to play, the screen changes to a display of a board holding a dial in front and an board. The board shows the moves around in the game, as you can see the dial holder is a hand over of the dial.

The dial is a large wheel with numbers 1 to 20 and a hand pointing to the number. The dial is a large wheel with numbers 1 to 20 and a hand pointing to the number.

The game is played on a board which is a purely numeric feature and you simply sit back and watch the computer throw its dice. As a simulation, game is not in the best of the best.

It is a simulation, game is not in the best of the best.

But, through the game, fun to play. It is a simulation, game is not in the best of the best.

**Timezone**  
Name: *Timezone*  
Field: *Timezone*  
Support: *Timezone*  
Developer: *Timezone*  
Publisher: *Timezone*  
Release Date: *Timezone*  
Price: *Timezone*



1000

Though the action in the game is fast and furious, it is worth a look in the screen display. In three days of scrolling, various light screens and 3D scenes. Code

I never admit to never having had a good night's sleep. However, if the general, I am not going to be as difficult as it is with these, non-political cases. I don't really know.

These have maintained the manual version of the *File-Journaler*, which if my performance is anything to go by, will take upon some time. Code Monkey has a manual, some stuff, and

one of those "good is more, go" types of games. It's good fun, the two players have a go with, no. Perhaps a little good display that starts to come to look that good.

**Townline**  
 Fair Inn Bed & Breakfast  
 44000 E. Highway 100, Box 10  
 Fair, N. Idaho 83421  
 208/338-2222

100



There are some games that the computer is just right for while others are strictly destroyed by being transferred onto the more wizard-like *Ultima* and *Ball Pool* seems to be of these conversions. I don't live up to its

depends upon events which you feel the man had with the law.

Whether from Top Gun allows you to select where you would like to send the main ball and at what strength, but that's all. In fact, estimates of the strength of the shot are "no more accurate than a guess."

1. turn on, place one ball in  
 one pocket, one player  
 pocket the ball on other one  
 player put each ball in its  
 correct pocket two players  
 put each ball in a pocket  
 and move the pocket, two  
 players move p. a. one  
 player to go for the angle  
 below the other hit w  
 player go go go

[illegible]



It's not like driving a car through a holiday street in downtown Boston, which, racing its powerful engines through tight traffic and horns going, back into your car in order to drive as fast as possible through a traffic snarl hovering over the main entrance. Fighting it out with other street pumps and us on their Coast. Motors have put the job far too big a strain on their little engines.

Your job description as a start man in Code Masters Linux spec will be to manage your way through all of the challenges mentioned above within a certain time limit. You have just four tasks in each session and you must complete each in turn. Of course if you fail in your task, then you'll need to start all over again, so be clear in your mind!

**Know Your Materials.** In all these last action papers that make you feel like you feel one more go will give you a better sense. It perhaps you feel sure that you've mastered the speed boat racing and one more go will see you through it and onto the next stage of the race.

If you're the sort of person who can't stand finding out that the time is 3 am and you're still playing that stupid game, but you'll have just one more go, then start here this program. On the other hand, it's a must for all addicts of that awful arcade music.

**For Sale**  
 Fully Equipped Sewer Pump  
 Supplier Code Waters PD  
 Box 6 Leesport Pa  
 Dispatch (717) 838 Tel  
 800-440-1212



**T**ime to get the old broadsword hiked and ready. Camille Waverro is a graphic advocate of some complaints, but there's still room for the buck and shag if you can make it through, in a matter of getting the feet Mags. There's the Fast Mags, Kings. Not easy, and reasonable for a budget price.

**Jeans:**  
Tuff Camille Waverro  
Saggy Waverro, 24  
Tennis Fast Racer  
Steel London W/1 202  
Pace 11 99/100

**Tooling:**  
Tuff Camber Hammer  
Supplier: Walworth, 24  
Limes Road, Pudding  
Road, London W11 2NR  
Phone: 0181 6061



# Pilot - a Programmer's Language

*If you're interested in educational programming but have yet to find a suitable language, Pilot could be the answer.*

*by*

*By Evelyn Mills*

**P**ilot is a language designed for those who are interested in educational programming. The particular version (Commodore release 4.2) is a compact version, easy to use and has approximately 10,000 bytes available for programming.

The design of Pilot is designed as a simple "question and answer" program which is useful in that the user "gets" apart from all the important design elements there are none.

- a simplified graphics screen with full colour control
- random screen drawing and handling procedures
- the ability to change all keyboard characters to any design you wish
- sound programming available

First of all the simplicity of design programming should be noted. There are three modes of operation - EDIT, COMMAND and IMMEDIATE (the latter is essential for producing whole listings).



In EDIT mode (press E key) the user may interact directly in the form of a hardwired screen prompt output and is limited however to the normal column screen width. Most important, in EDIT mode, is the facility to delete text, copy text and generally make amendments as required.

## Real Time Programming

To get going with real time programming, begin in COMMAND mode (press G key) and use the comparison with the EDIT mode in the PILOT modules for instructions. Consider a simple example such as:

T8 What is your name please?

A

T Hello, PSB press RETURN to continue

A

15. (a) In using personal requests (which is subsequently followed by a 10-min. wait) (p1) as well as open 5-min. wait (p2) the waiters are not helped by the wait. New lines are simply provided by a robot which is a kind communication system.

Naturally, then, must be flexible in control while programming otherwise, the dialogue will be nothing more than meaningless PLOU pseudo simple steps for the and has restructured the most advanced features of Basic, into a simple format. A sample of such Basic now is follow, each instruction must be followed by a colon, a line number here.

- FF - problem solving
- D - dimension
- W - wall
- M - match
- J - jump instructions
- compar to jump a li
- within the program
- C - correct answer of a ques

Variables and constants which cannot be manipulated from user programs are associated with the machine type and jump commands. U (user) calls instructions which are associated with user programs.

### When Numerical Answers are Required

Questions which require numerical answers are streamlined to a high degree: e.g.  $\lambda_1 = 0.814990$  returns a number between 0 and 99. All the subroutines handled by the C24 are portable and are computed using an AMD architecture which automatically

The manual is full of examples which will take you through every aspect of Pascal programming, speedily and efficiently. You have here a well-presented and, for reference (and fun!) well worth using, structured course.

Age Group	No (%)
18-24	~5
25-34	~5
35-44	~5
45-54	~5
55-64	~10
65+	~5

Turning to the LORAPHEX code we start by pointing to `E` followed by the negative field definition. The `s` and `i` vars are "14 to 19" respectively, the `l` represents the line, and the `l` stage has 1 based numbering. Finally, the `var` section contains

decided by the need (P) followed by the given to treatment (D). Therefore a change in the command could be implemented as:

© 1999 by John Wiley & Sons, Inc.

The graphs here may be moved to another part location using an `M` sub in line state by string with an `R` command both of these commands need of course, the request is : `loc:move`.

In combination with this, the test curves may be positioned exactly where required; the curves may be split using an S command and linked to Frame or Window Drawing for maximum display effect. The full colour range of the Cid is available with single letter combinations for border (X), background (B) and foreground (F) colours. The FILL command (F) is used to fill a block with a background colour by specifying the digitised colour number in the colour master file (i.e. 01, 02, 03, 04, 05, 06, 07, 08).

First, again looking into the design, allowing time to design sprays and screens and to make them allowed additional sample investigations. The BIT-PAT188 [18] is used to define wave sprays on a 21 x 24 grid, using a dot to create the grid and an 'x' to indicate the pattern. The sprays should be given a number 1-31. To start the program type: 1 2 2 and go ahead with wave pattern 10 (spray number 1). Having done so, the screen displays, for a second, an empty 60 x 60 grid and a legend code (Fig. 1). An example is shown in Fig. 2.

SEI 09-104-108  
Rev. 10/09

The spritz will run, by apparent colored black at location 100-100-on

color change.

Agnes may be expanded horizontally or vertically, expanded to normal size or used in multi-column mode [M] with three possible columns in any one Agnes. Agnes full illustrations are given in the manual combined with the sample commands for various states around the screen.

Chaudhary, M. and M. S. Khan

There are many applications to which specialized software would be useful, particularly in foreign language study, where it can display dialogues.

character may be obtained by using the NCHAR function and the following by a name or corresponding to the standard ASCII code of the letter and (12-12). The code is 65 for example, so 65 will allow recognition of the letter. Recognition is done on an 8-bit system so much the same manner as before.

© 2005 Blackwell Publishing Ltd

This enhancement is very useful in detecting malfunctions in a dialogue, or examining the data can only be regarded as a supplement of and will not give you any degree of statistical analysis. However, both the end user and the system can be programmed with ease to good effect and within the limits of single enough for use. Both single and two block operations are available as are ring modulation and such operations. Volunteering and character of many attack, close, distant, release, are all there and a second release is available on the disk, to allow a more sophisticated work. The third variation. Many more values are, based on the commands.

There are more advanced features in Pict which show how far it gets beyond multiple choice questions with selected answers (just under ENHANCE), while the ESCAPE mode allows special features to be built up within a program not called by the system at all later. A good example is given of how to create a calculator which may be called at any point within a program.

In this event the user will be asked a long program and two cell members, there is a LINK instruction (L PART) to be found in the second part (and subsequent parts if specified subsequently). Naturally, the previous program is then available in the cell registers should be subsequently used.

The normal history of sample loading printer output are all available as a "Raw Gish" version of Pictas on the disk, which means that this program may not be listed edited or printed. The disk, however, contains demo which listed will show you how to handle some graphics, edit screens and programmable characters.

For the program, Pilot is an easy-to-use, interactive system which can be self recommended, run all the code and a foreign client Database Connection Ltd.

# Tank Attack



With videogames becoming more and more sophisticated, it's good to see that our company, CDS, has not forgotten the beginner. What is doubly essential is that their latest title, *Tank Attack*, makes a winner behind a computer-modified weapon.

The scenario is simple. Four neighboring countries—Armenia, Kazakhstan, Europe and Colombia—declare war on each other—well, on two of the others actually, but there are two first advantages. Resources are limited to armored vehicles, and the weapon is to blow holes in your enemy's command post before it does likewise to you. The game can be played by two or four players. That is no rule option.

The board itself is thought up on a board by moving little plastic tanks around a hex grid. Each country has two light and medium armored units as well as two light medium and heavy tanks. The pieces are cunningly designed so that you can't tell the strength of an opposing piece when it's on the board—you can only find out by attacking it. The pieces can be placed anywhere you want within the boundaries of your country. It's up to you to come up with the right balance of attack and defense.

How far you move each turn is determined by the computer taking into account the weather for the day, and several other random factors. The number of hours that you can move can be split between as many vehicles as you see fit. There are a few turn-of-the-minute and a bonus for unaccounted cars in open space.

If you want to attack, you must inform the computer how far away from your target, you are, and then discuss what power vehicle you have. Your move does likewise, and the computer then works out the result of the battle. At this stage, you get to see the battle through a stylized pair of binoculars on the screen. (Incidentally, I wonder

if the price designed title even actually looked through binoculars. The sideways figure of eight view is not what you get. Hollywood directors, please take note too.) The result of the battle is then displayed, effects ranging from total destruction to a vehicle being removed to a repair status.

That's really all there is to the game. The board looks attractive, and the on-screen commands are clear and simple to use, but then it's a huge butt who battles in the first place? The game would have worked just as well without the computer being there. All you need to do is determine the move factor manually, and look up the results of combat on a table.

If, however, the game had been entirely controlled by computer, it would have been truly excellent. All the problems associated with board videogames could have been eliminated at a stroke, and a host of other features included—a save game facility should something interrupt the battle, a solo option, better movement, deliberate unkill screen effects, and dare I say it, the elimination of cheating, or at least the urge to cheat. The idea of putting the computer to tell you how far you can move, and then making the player go and look up the rule book exactly where he can move, does seem to be something of a retrograde step.

*Tank Attack* is certainly well produced, but I feel that it is the wrong game for the combination of board and computer allies.

#### Tonkolider

Title: *Tank Attack* Supplier: CDS CDS House, Bristol Rd, Dorchester DT97 4AD Price: £12.99 (incl. £4.99 incl.)

## “THE MAIN EVENT”

DEPT (YC)-61 STAFFORD STREET, HANLEY, STOKE-ON-TRENT,  
STAFFORDSHIRE ST1 1LW  
24 HOUR ANSWER SERVICE TEL: 0783 281544

[illegible]

Please email [RSVP@chicagoandmichigan.com](mailto:RSVP@chicagoandmichigan.com) and bring something to THE MAIN EVENT

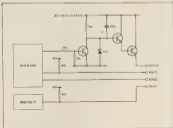
To help maintain our low prices we would be obliged if you could include a firm class stamp with your order, although this is NOT compulsory. Free Post & Packman. Overseas orders add the sea rate.

We apologize for any inconvenience or confusion caused during the move. The Island Express is a division of Redmond City.

# Tech Troubles

*Don't get stuck in a rut  
Let our agony uncle  
help you out*

**By Tim Arnot**



Connection circuit between

two segments of pulse information for moving an information. While the reader of statements, suitable for the job of looking and saving programs for your application of digitally sounds, it is a unity action.

The Commodore cassette mechanism basically records information as a series of digital pulses of different length. A series of these lengths is then used to determine whether a bit is a 0 or 1. Thus, a 1 is represented by a short pulse followed by a medium pulse; a 0 is medium, followed by a short, and the start of a new word is a long followed by medium. I have included a brief schematic of the cassette read/write logic so you can see what happens. Notice that the READ signal is fed directly into the FLAG input of CLAR (SDAC00-SDCTT) which is basically an edge sensitive interrupt input.

In order to sample sounds, you will need to supply a device called an Analog-to-Digital Converter (ADC). This basically takes a voltage of the sound (represented in

bits) and converts it into a number for entry to the computer. The quality of the sampled sound is dependent on the speed at which you take these samples. For example, compact disc, which is essentially sampled sound, has a sample rate of 28,000 samples per second.

To show you how to construct a sampler based on the ADC, it is built beyond the scope of this column, but you can purchase some ready-made from a number of vendors, most of whom advertise regularly in this magazine. Cost would have the price of \$50, but that will also include software to do this.

Dear Tim:

I am trying to get my Commodore 64 to sample sounds from the cassette, and have implemented a circuit with an oscilloscope to read the sound coming in. I would like to know how to do this, and could be very grateful if you could assist me. Some information on this subject or put on article in your magazine. I will be a complete beginner and would appreciate and I feel forward to your reply.

**SD Moore, Chesham**

Hi Neil:

Thanks for including an S.A.B. As a general point, after you have read the Your Commodore efforts, it can be up to a month before they reach me for reply. It will then be another month before they appear as print. As if you want a reply quickly, I will always send a copy of my reply as an S.A.B.

Unfortunately, (as you already) the Commodore cassette system is not a digital and when pulses on the

Dear Tim:

I have a Commodore Plus/4 system and have a question which I hope you can help me with. I am doing some programming in the "new part" - rather than BASIC, or assembly language.

I am just concerned with the correct word for the Commodore 64 to

*For the reader to read all this, it's the  
most boring and uninteresting  
approach.*

*Steve Jones, Gillingham*

**Hi Steve**

Unfortunately, I have had news for you - the Plus4 doesn't have a test point - at least not in the sense that the PET 64 and 128 have test ports.

To explain the PET 64 machines have an edge connector at the rear labelled Parallel user port which provides a general bi-directional I/O port that can be used for general interfacing applications. And indeed we've a range of peripherals that use this port. From relative adaptors to Commodore printer drivers. However Commodore really intended to implement this port as an RS232 interface using a 6551 Asynchronous Communications Chip. Due to an ordering error up, they couldn't get the IC, and as the parallel port was using RS232 was implemented as follows:

In the UK, if we want RS232, we normally buy an interface that plugs into the cartridge port or serial bus and connects via RS232 and some dummy drivers. This is because the built-in port can't handle more than about 500 baud reliably. But in the States all these modems and things plug into the built-in RS232.

And as for the Plus4, to give you Commodore finally got their act together and produced a real RS232 port using the proper hardware running at decent speeds (up to 300 baud). This is what you have on the edge connector at the back of the machine. It works in the same as the 64 (from the RS232 viewpoint) but the 64 uses a peripheral that plugs directly in. The right way to be of course, except to UK users who want the parallel port. And unfortunately 1% of the users are Americans who want RS232.

**Dear Tim**

*There is no way to get a Commodore 64 to run a program for a third run and then for a Plus4.*

I agree that there will not run on the 64 but that is not what the Plus4 commands were to do. After the user loads the Plus4 but how do I get them to load and run? I don't want to do it every time just for RS232 but

*how do I do this and how should it be?*  
*W J King, Worthington*

**Hi Mr King**

From the BASIC point of view the essential difference between a 64 and a Plus4 is the amount of memory available. The additional memory graphics and disk keywords of the Plus4 and the same location for BASIC text.

Of these, those the memory aspect can be discounted since you are writing on the machine with less memory to transfer onto the machine with more.

The sort of BASIC installed is not really important, as all BASIC programs on the 64 and Plus4 (and C16 and 128 machines) are designed to relocate themselves to wherever the sort of BASIC might be. The problem of this is that LOAD "PROGRAM" will work on one machine except the PET. As to where the start of BASIC sits, on the 64 it is at location \$B001 and on the Plus4 it can be at one of two places depending on whether you have a graphics screen or not. These two locations are \$1000 (without graphics screen) and \$2000 (with graphics screen).

When you type a line of BASIC and press the return key, the computer stores what you have typed for words that it recognizes and converts them into one or two byte tokens. This was originally done to save memory - each token takes up much less space than the word itself. Any words that it does not recognize are assumed to be variables and so are left alone.

When you run the program, these tokens are entered into a lookup table which points to the version of BASIC that performs the function was are asking the computer to do. If the keyword has not been tokenized (for instance typing FLASH on the 64) it cannot find the entry in the lookup table and so will crash. Similarly BASIC 1's tokens will fall off the end of the 64's lookup table and it will also crash.

One further point regarding transfer of programs from one machine to the other: the cassette system used by the Plus4 and C16 is incompatible with that used in all other Commodore computers so programs can only be transferred through the use of disk media.

**Dear Tim**

*I am writing to you because I think your magazine is one of the most enjoyable point of view in Commodore computing that will be such a C64 and Commodore 128.*

I have a very important question to ask. I have a wish to be both magazines: where I wish 3 & 4 I read had an reply to I hope was worth you will be to others. I bought the C64 last Christmas and I have been getting on well. I will not be used I found out that I was buying the country is so to C64 (North America). To my question is not it would ever more? Do I need a full of code? If so, what? I am using a 3.1 I will I need a monitor? Well a monitor just an ordinary one on a Plus4 is probably worth good money.

P.S. what do you think of the book quality? Printed by the Silver Reed Culture Plus Graph. Could you tell me if there is any compatible software for the book graphs?

*Rodriguez Arroyo, Madrid*

**Hi Rodriguez**

OK, the first bit first - you don't need any special cables, neither will you need to use a monitor - a TV screen acceptable regardless of the country you are in. As for the rest of your question I had to do some digging. In the UK, and in most of Europe, we use a TV system called PAL, which stands for Phase Alternating Line. Additionally, our TV screens are off scan lines' making up the video picture. Chile and most of South America has adopted a different TV system called NTSC, which stands for North American Television Standards Committee. Their TV screens has 32 scan lines on the screen. The result is that the two are mutually incompatible and you will have to sell your 64 system and buy another one when you get there.

The major voltages in Chile are also different to ours, and so the 64 would not work even if you got your PAL television with you (which could have been as cheap although you would have been able to watch local TV on it).

The quality of your general request, good! It is a great compatible - it should be able to run most of the general graphics packages that are around at the moment.



# Sprite Library

*More tips on Sprite manipulation on the Commodore 64*

*by Mike Benn*

**T**his month we get out the geometry set. All the sprite definitions are based on single sprites and use lines. The table lists the variations on the month's theme with a mixture of movement sequences and single shapes.

Load in the basic loader and run it, if error free, the program will automatically open itself as a block of data. If you reload that data in the future, remember to add a 1 after the device number. The data is saved in the following location: \$3000-\$37FF.

The sprites run from 194 to 223 in a compressed to avoid the size

\$3000, traditionally an aside for redefined character graphics, and to avoid the need of typing in line after line of data.

If only one or two sprites are required then author's format (sprite block No. 160) \*40 + 194 = the data line number at which that sprite block's data starts. Remember to type in the following three lines of data, and alter the variable BL to the number of data lines you have in your finished program, last 1.

The small basic program GRIM DISPLAY, will variously animate the sprites in both row expanded and expanded forms on the screen simultaneously. To load on any sprite, enter the same number for Start and End.

Any sprite Editor program will enable you to change and adapt the individual sprites to your own requirements.

## GETTING IT ALL IN

Type in the basic loader as published and save it—don't run it or it will self-destruct. Before running the loader program, you'll need to start the computer and type directly in the following:

POKE\$A,\$POKE\$B:POKE\$C,\$POKE\$D:NEW

and press return. This will load the computer into believing that the basic now starts at \$4000 instead of \$0001.



## TABLE (Geometric — Hires)

A6 - A3 / 158	153 Growing triangle	CT / 207	Inverted pentagon
A3 - A6 / 163	166 Spreading triangle	D8 - 84 / 208	213 Growing hexagon
A7 - A0 / 167	171 Growing square	D0 / 213	Twisted hexagon
A9 - B1 / 171	179 Rotating square X axis	D6 - 214	6 pointed star
B3 - B6 / 179	187 Rotating square X axis	D0 / 215	Twisted six pointed star
B8 - B0 / 187	189 Turning square	D4 - 216	Cube
B0 - C3 / 189	195 Rotating diamond X axis	D4 - D0 / 217	223 Growing octagon pattern
C3 - C8 / 195	201 Rotating diamond Y axis	D0 / 223	Octagon
C4 - C6 / 201	206 Growing pentagon		



# Power Play Hockey

**W**hen it comes to single-screened computer hockey games, *Power Play Hockey* is a real winner. It's a fast-paced, action-packed, and fun game that's easy to learn and hard to master.

The game is set in a 3-on-3 power play situation, which is a common occurrence in hockey. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

The game is played on a standard ice rink, and the players are represented by small, stylized figures. The game is played on a standard ice rink, and the players are represented by small, stylized figures.

## Verdict:

*Power Play Hockey* is a fast-paced, action-packed, and fun game that's easy to learn and hard to master. It's a real winner.

# Personal File Database

*Avoid elaborate and time-consuming editing procedures with this word processing/database system*

**By Eric Randall**

This utility is a cross between a word processor and a database system. It's aimed at the user who needs to maintain files of text, and be able to readily update and search the material, without the need to learn and remember elaborate editing procedures. To achieve this, it uses the C64's normal screen editing functions: with one or two more added, and all the conventions needed to use the system are displayed (except when setting up an editing page of text).

The data is held in pages, containing complete screens of information. Each may be recalled and updated as often as required, using the function keys as explained on the main display. New pages may be inserted or old ones deleted, and the entire file in memory can be printed. If when editing a screen, you need to refresh your memory, or which function key to use, press cursor to

save what you've already typed in. The main display will appear and, after checking the conventions, recall the page you were editing by using the F1 key.

Having set up a file of data, it may be saved to disk. To simplify the disk housekeeping, the program takes care of naming files, and you're asked if you want to overwrite an existing file. The file names start with "Pfile" and end with a number in the range 0 to 9. All the user needs to remember is which number is used for a particular file.

For the technically minded, the data is held as screen codes, not ASCII codes. If you want to control the scope of the system, this must be taken into account. Furthermore, the data are compressed before being to disk, so the disk file format differs from those displayed on the screen. The areas of memory used by the system are as follows:

Basic Program	\$6000	\$1100
Housekeeping data	\$6400	\$64FF
Storage for screen pages	\$6500	\$B81F
Area for packing and unpacking data prior to storage on disk	\$BC00	— \$BFFF
Maximum	\$C000	\$C400



For those wishing to experiment with and extend the system, the entry points to various machine code routines called from the main BASIC program may be useful. Most of these

routines require parameters which are taken from various locations between \$6400 to \$64FF. These are set up in the main program, and the references are as follows:

SY\$4933 (\$C17F) turn off interrupt routine  
SY\$4932 (\$C000) - store screen

#### UPDATE PAGE

Lines 708-728 SY\$4934 (\$C070) - display stored page  
SY\$4931 (\$C160) - turn on interrupt routine  
SY\$4935 (\$C17F) turn off interrupt routine

#### LOADING FILES

Lines 708-728 SY\$4933 (\$C16F) load a file  
SY\$4934 (\$C44E) unpack data

#### SAVING FILES

Lines 59-218 SY\$4935 (\$C1A19) - pack data  
SY\$4936 (\$C090) save a file  
SY\$4934 (\$C0C4) - overwrite existing file

#### CREATING NEW PAGE OF DATA

Lines 500-557 SY\$4931 (\$C16F) turn on interrupt routine to scan for function keys which tabulate up and down and which insert and delete lines

#### PRINT FILE IN MEMORY

Lines 600-660 SY\$4934 (\$C0C7) Print file

#### DELETE A PAGE

Lines 600-665 SY\$4935 (\$C1FC) delete a page

#### INSERT A PAGE

Lines 600-665 SY\$4930 (\$C150) insert page

#### USING THE SYSTEM

To start you off using the system, a data file is included on the disk as file number 4. This contains the headings of personal information for you to fill in the details

# Introduction to

# Graphics on the

*by Allen Webb*



*The first of a two-part series examining the considerable graphics potential of the Commodore 64*

**F**or many new owners of the Commodore 64, playing games soon becomes inadequate, and they turn instead to programming and try to evaluate the graphics effects that the game writers achieve. The machine's manual will help you a little, but it makes no attempt to get to grips with the vast considerable graphics capabilities of the machine. In this series, I'll give you a taste of what you can achieve with Basic. You must realise that using Basic will limit your potential, but you will nevertheless be able to get some excellent results.

The heart of the graphics power of the 64 is the VIC chip, which has a collection of memory mapped registers which handle graphics. It's a curious fact of life that Basic doesn't support graphics in any way. Instead you'll have to learn a collection of POKE commands to manipulate the VIC registers. Table 1 gives a summary of the registers and their functions.

TABLE 1

Memory Address	Function of register
53280-53283	Sprite position registers
53288	Control Register 1
53289	Sprite display enable
53290	Control Register 2
53291	Sprite Vertical Repeat
53292	Memory Control Register
53293	Interrupt Flag Register
53295	Sprite Priority Register
53296	Sprite colour mask
53297	Sprite horizontal repeat
53298	Sprite collision detect
53299	Sprite colour detect
53300	Border colour
53301	Screen colour
53304-53306	Background colours
53307-53308	Sprite multi-colours
53309-53310	Sprite colours

In these articles I will explain how to track these registers to good effect.

First, I must discuss how the machine manages memory. The Commodore 64, as indicated by its name, has 64 kilobytes of memory. This can be organised as a sequence of 8192 8-bit bytes with one address each. Each byte (or memory location) is numbered, starting at 0 and continuing to 8191. The number of the last is called its address.

The machine uses various chunks of memory for various purposes. Due to an 8000h-based address, the 64 uses the same areas of memory for its various functions. You can, however, change things around if you wish. The VIC chip can address only 16K of memory at any one time. The 64 has four "banks" of 16K available for use and these are numbered as shown in Table 2.

TABLE 2

Bank Number	Block of memory	Z
0	0-16384	0
1	16384-32768	1
2	32768-49151	2
3	49152-65535	3

When you turn on the 64, bank 0 is automatically selected. This means that the VIC chip expects to find the screen memory and the character set designs in this area. You can, if you wish, change the bank by changing a register in another important chip, the CIA (number 1). The bottom two bits of location 56276 determine the memory bank in use. To change the bank, you use the following two lines:

```
POKE 56276:PEEK(56276):OR 1
POKE 56276:(POKE(56276):AND
32):OR 2
```

Using some banks will give a screen full of garbage. I will mention this later, but for now, we forget all about it for now.

The display uses a block of memory to hold its contents. The display consists of 24 lines of 40 characters, requiring 1000 bytes. In the default 64, the screen memory starts at memory location 1824 and continues to location 2823. In order to provide colour, each screen location has a corresponding colour memory location. This occupies the 1000 bytes starting at 53296. The colour memory is always at this address. Try the following program to see how it works:

```
10 FOR I=0 TO 999
20 POKE 1824+I
30 POKE 53296+I:RND(1)*16
40 NEXT I
```

The machine demands to know what shape the characters are and this information must be held in the screen memory bank as well. Because the 64 is an 8-bit machine, it allows a maximum of 256 characters. Each character requires 8 bytes of memory to hold its shape data. Hence, the full character set requires 2048 bytes. Since there are two full character sets available, a total of 4096 bytes are used. In the default system, the characters are held in a Read Only Memory, and a found in the block 4096 to 4191. Because this data is held in this way, it occupies the same space as Random Access Memory, so that you don't lose any memory. Given these facts, Table 3 shows a general memory map.

TABLE 3

Memory Address	Use
0-8000	Used by system
1824-32768	Screen memory
32840-32847	Sprite position pointers
32848-40959	Used for BA/VIC programs
53296-53299	VIC registers
53296-56295	Colour memory

The parts of memory not described are occupied by the system ROMs and other chips. The position of the screen memory and the characters are determined by the contents of the memory control register in the VIC chip (53276). More on that in a moment.

When you put a value in a screen memory location, the system translates that value into a specific character pattern in the character memory. A value of 0, for example, uses the first pattern (a group of eight bytes). This pattern represents @. A value of one uses the second pattern and so on. The following program shows the full character set in POKEing the full range of values into the screen memory.

```
10 FOR I=0 TO 255
20 POKE 1824+I
30 NEXT I
```

The next step is to place a character at a specific place on the screen. If R is the row and C the column, the

current address is given by

12100	12
14100	14

$1024+(R-1)*40+(C-1)$

Where R is in the range one to 25 and C is in the range one to 40.

The following example uses the character set as a function to move an asterisk diagonally across the screen.

```
10 DEF FNA(R)=1024+(R-1)*40+(C-1)
20 FOR R=1 TO 20
30 C=0
40 POKE FNA(R),42:FOR D=1 TO 30
50 NEXT D
50 POKE FNA(R),12:FOR D=1 TO 30
60 NEXT D
60 NEXT R
```

The routine first puts an asterisk in each point (character 42) followed by a space (character 12). The two loops in lines 40 and 50 are simply devices to give a better effect. While making some progress, the effect is a little boring. What's needed are custom-designed characters, but how can we achieve that? As I have already described the normal character set is stored in a ROM and what we want is a character set in RAM. The secret is to redirect the pointer to the character set to a handy bit of unused RAM. The memory control register allows this. The bottom four bits of this register decide the effect position of the character set in any given bank. Take a look at Table 4.

TABLE 4

Memory Offset	2
0	0
2040	2
4080	4
6140	6
8192	8
10240	10

The character set pointer is changed by using

```
POKE 53273,(PEEK(53273)&AN-10240)OR 2
```

To find the start address of the character set, not simply add the offset to the start address of the bank (given in table 1). Let us consider how a character is designed. Here is a sample character.

```
00010000=24
00111100=60
01100110=102
01111100=120
01100110=162
01100110=182
01100110=182
01100110=182
00000000=0
```

The main represent visible data, and the data represent spares. Each pattern can be converted to an eight bit binary number by replacing stars by 1 and dots by 0. These numbers are then converted to decimal, and the resulting eight numbers are those used in the pattern table. Here is an example of a redefined character.

```
00000000=0
00111100=60
01111100=120
01001100=162
01001100=162
01111100=120
00111100=60
00000000=0
```

The following example uses this character plus one other to produce stunning animation. Gasp!

```
10 POKE 53273,(PEEK(53273)
&10240)OR 14
20 DATA 0,60,120,162,162,120,60,0
30 DATA 0,0,0,24,0,0,0
40 FOR D=0 TO 15:READ X:POKE
10736+X,X:NEXT X
50 FOR D=0 TO 7:POKE
14076+17*D,X:NEXT D
60 DEF FNA(R)=1024+(R-1)*40+(C-1)
```

```
70 FOR R=1 TO 30:FOR D=1
80 POKE FNA(R),X:GOSUB 200
90 POKE FNA(R),1:GOSUB 200
100 POKE FNA(R),12:GOSUB 200
110 NEXT R
120 END
200 FOR D=1 TO 30:NEXT D
RETURN
```

In this example I have placed the start of the new character set in 14176. Line 40 reads the new character data and puts it into the character data table. Line 50 holds the design described earlier. Line 60 sets the custom character (12) to a space. The rest of the characters are garbage, since we haven't set them to anything intended. If you want to return to the normal character set, the quick and dirty way is to press RUN/STOP and RESET together.

The normal display comprises two colours: the background and the foreground – as Sinclair Spectrum users, paper and ink, and some border colour. These colours can be easily altered by putting the colour value (five 15 bits) into the relevant register. The registers are:

Border	53280
Background	53281
Foreground	540

Since these registers use only the bottom 4 bits of the stored value, any value greater than 15 will simply result in cycling the colour sequence. In general, you should ensure that the value stored in any VIC register is accurately controlled since an incorrect value will at best give odd results and at worst a nasty crash – be warned!

In the default or high resolution mode, 4 bit points in the screen is displayed in foreground colour, and an exact point in background colour. While this gives the greatest clarity, it is not best suited to pretty graphics. So we enter medium-resolution mode, which can give you up to four colours in any one character. The proviso is that the horizontal resolution is reduced from eight dots per character to four. The reason for this is that each pair of dots in the character pattern determines which colour is used. The colours are chosen as shown in Table 5.

TABLE 5

Colour Source	Bit Pair
background 53281	00
background 1 53282	01
background 2 53283	10
low 1 bits of colour RAM	11

Let us consider one bit of a pattern.

```
"" "" - 00011011
```

In high resolution mode, this would look like a pair of dots. If we split it into four pairs of bits we get another image:

```
00011011 - 00 01 10 11
```

Using Table 1, we can see that in multicolour mode, the first pair would appear as the background colour, the second pair would appear as a dot coloured as background one, the third as background two and the last as the colour red. Because only the bottom 3 bits of the colour run is used, the first pair can only use the first eight colours. The other bits is used to toggle multicolour mode for the particular character. You see, if the fourth bit is set, the character is set to multicolour; if it is not to high resolution. In practical terms you simply add eight to the colour value for the image taught to memory. This allows you to mix both modes on screen. To turn on multicolour mode you use

```
POKE $3270,PEEK($3270) OR 16
```

and to turn it off

```
POKE $3270,PEEK($3270) AND 239
```

The next example sets up a black screen and puts a multicolour character next to the same character in high resolution mode:

```
10 POKE $3280,0 POKE $3281,0
20 POKE $3282,0 POKE $3283,0
30 POKE $3290,PEEK($3270) OR 16
40 POKE $3291,PEEK($3270)
50 POKE $3296,1 POKE $3297,0
```

The lines 10 and 20 set up the colour registers. Line 30 turns on multicolour mode and line 40 puts an A in the top left two screen positions. The first POKE in line 50 clears the fourth bit of that particular colour memory location, thereby setting the colour to white, and setting the character to high resolution. The second POKE sets bit 4 thereby setting the colour to white and setting the character to multicolour.

Trying to draw multicolour characters "by hand" is both tedious and

difficult - if you want to get good results, it would be worth your while investing in a good quality character designer.

If all of this is not enough, there is one more character mode - extended background mode. In this mode you have a high resolution character, but with a choice of four background colours. This time the system gets the colour chosen from the screen contents. In simple terms, the background colour depends on the character on the screen, and the background colour depends on the colour memory. Table 4 shows how.

TABLE 6

Colour Register	Character range
\$3281	0-63
\$3282	64-127
\$3283	128-191
\$3284	192-255

To turn on extended background you use

```
POKE $3285,PEEK($3285) OR 64
```

and to turn it off

```
POKE $3285,PEEK($3285) AND 161
```

Here is a simple example:

```
10 POKE $3280,0 POKE $3281,0
20 POKE $461,PRINT CHR$(167)
30 POKE $3284,1 POKE $3285,12
POKE $3283,12
40 POKE $3285,PEEK($3285) OR 64
45 FOR P=0 TO 255
50 POKE $3294+P,1
60 NEXT P
```

This is simply a repeat of an earlier program. Here I have used PRINT CHR\$(167) to clear the screen. On the screen this has the the colour RAM with the current background colour. By the way don't try to use extended background mode with multicolour mode; it won't work.

As I mentioned earlier, it's possible to release the block of memory used for the screen display by altering the relevant bits in the memory control register (\$3272). This time, it's bits 4 to 7. For most users, this option is not of great use, but it does offer the option of having several screens active at once with a simple means of switching between them. This could allow simultaneous equations and other effects.

In a manner analogous to the character set, the screen can occupy

a range of positions. This time, a row of six lines can, the screen can be placed in one of 16 positions. These positions are in increments of 16.

Table 7 shows the available options.

TABLE 7

Location of Screen (offset from start of bank)	X
0	0
1674	16
3246	32
4812	48
6386	64
7920	80
9444	96
10958	112
12472	128
13986	144
15500	160
17014	176
18528	192
20042	208
21556	224
23070	240

The first step is to change the pointer to the screen with

```
POKE $3271,PEEK($3271) AND 15
```

As before, the start address offset is added to the start address of the current bank to get the absolute screen address. We if you intend to use BASIC, you must also tell the editor where the screen is. A pointer to the page occupied by the screen is held on location 645. This must be altered. You get the page by obtaining the absolute address of the start of screen memory (as described on the last paragraph) and dividing it by 256. The 6502 micro-processor handles memory in pages of 256 bytes. The result is POK into location 645.

I must apologise to beginners for the large amount of theory in this part. Unfortunately it is unavoidable, and in any event, I want this short series to become a simple reference guide to which you can refer when necessary.

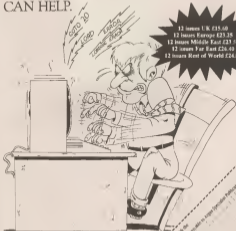
Well, that's about it for now. With the building blocks developed here, you should be able to start developing some interesting graphics of your own. Next time, I'll be looking at bit-mapped graphics and spaces. I will also include a small package of machine code routines which will ease the use of the VIC chip.

TRYING TO USE YOUR COMPUTER?...

YOUR

# COMMODORE

CAN HELP.



Send this form with your remittance to:  
**INFONEET LTD**, 5, River Park Estate,  
Berkhamsted, Herts HP4 1HL



Please type or stamp in the following information to help us process your order  
I enclose my remittance for £  
or please  
by cheque  
Name (please print)  
ADDRESS  
Postcode  
Signature  
Date  
Name of your company  
or individual

# Relative File Programming

The new *Journal of  
 American Studies* has  
 placed it in the category  
 of "general interest"  
 and has put up the  
 following notice:

**T**he authors of the paper, David L. Foray and Robert C. Anderson, are with the University of Maryland System, College Park, Maryland. Foray is also with the University of California, San Diego, La Jolla, California. Anderson is also with the University of California, San Diego, La Jolla, California.

1. *Journal of Management Education*, 2000, 24(1), 10-19.  
 2. *Journal of Management Education*, 2000, 24(1), 20-31.  
 3. *Journal of Management Education*, 2000, 24(1), 32-43.  
 4. *Journal of Management Education*, 2000, 24(1), 44-55.

At this point, the project is being financed by a consortium of 10 companies, including the two main contractors, and is expected to cost about £100 million. The consortium is led by the two main contractors, who are also the main contractors for the other two projects. The consortium is expected to be formed by the end of the year.



It is very important to understand that the "best" treatment is the one that is best for the patient. There is no one "best" treatment for all patients. The best treatment is the one that is best for the patient. The best treatment is the one that is best for the patient.

There is considerable evidence to suggest that the use of a structured format for the interview is associated with the use of a structured format for the analysis of the data. This is particularly true for the use of a structured format for the analysis of the data. This is particularly true for the use of a structured format for the analysis of the data.

# Mailbag

Your chance to have your say



When I look at my 128 and my Amiga I am reminded strongly of the old fable of the Grasshopper and the Ant. The Grasshopper played all summer long, leaping and punting all over the meadow, shoving off, but never doing anything really useful. The Ant on the other hand toiled steadily away, doing all the real work and laying up stores. When winter came the poor old Grasshopper had no reserves so he perished. At least the Amiga has made it through the winter, and done enough on the value trail to make sure it will prosper, but while it has proven the machine for the future, I must say that the 128 is that of the here and now.

I am not knocking the Amiga. I think it is a wonderful machine. If your main interest is in video graphics then the Amiga is your only choice. It is what I had hoped the C64 would be like when I bought it nearly five years ago. I remember, however, leaping around the computer shops trying to find a decent program, and am amazed at the things that are now being done with a C64. Even the designers hadn't any idea how powerful it would prove to be. The Amiga is in the position now that the C64 was five years ago full of potential and potential, but with no really great programs around to exploit it. I only hope I won't have to wait another five years for the

programmers to get to know the Amiga.

I started with a C64, but switched after an 80-column screen, so I took the plunge and bought a Commodore 128 and high-resolution colour monitor. I have never regretted it. Despite the scarcity of programs written for 128 mode there were enough to make the 128 a real work-horse, and in any case one could revert to C64 mode to play *Elite*, *Lander* and *World Class Chess* or *Bridge*.

But always there was the sheer weakness of the 128. It took on my wife's correspondence as seriously as a charitable organisation, did *Drawers*, *Dates* for the week, and then produced professional looking posters for both of them. Later I ventured into writing simple programs on the enhanced Basic that can be found as the programming language of *Superbase*. It is this feature, however, the draggling aspect of home computing, which frustrates me.

When the Amiga first arrived I was pleased out of my reach. I could only say those who could afford one I think the majority of 64, 128 owners felt the same. Then came the A500 and I decided that I ought to keep up with the times. I chose the relatively cheap one, by having an A500 and having my 1281 Monitor converted by Triplex of Isfield for only £25. It quickly the dedicated Amiga Monitor

for colour and definition, but it was, of course, still too small for the C128.

I have confidence that I was almost drowning over the new possibilities - a reasonably solid machine, and a very speedy disk drive. New technology would open the flood-gates and things would really take off. True, there weren't many programs available, and even games were twice the price, but time would turn that. The weakest noticeable difference was the graphics. No more chunky blocky screens. No more peering into a binocular screen trying to peer what chunk of block was the aim. I bought *Defender of the Crown* and marvelled at the wonderful detail. Then I took a flight round the *Voice of Liberty* and the set set in. Instead of the rounded body of the C64 (often floating past the side windows rather politely) I could see only a hard board cut-out. I was in wrong. Maybe my copy has a glady. But the first hint shadow of doubt had come in. The car was beginning to cloud over a little. I returned to *Callender of the Crown*. The scenery was as delightful as ever, but was the game-play a little weird? Even boring? Yes it was. Very! I bought *Landlord* for the Amiga. Quite frankly I preferred the multi-panels of courses on the C64. And it played just as well as on the Amiga. *Star Gladi* was quite good but I prefer *Elite* on the 64.

Ah, yes. But I must be

stuck better for the really productive work? Well, I have tried *Scrabble*, *Panasonic Dealings* and finally *Wordperfect*, but for a really user-friendly yet very powerful word-processor, which can be maintained with a touch of its magic, give me *Superbase 128* as its reliance on the C128 may turn. This I know that familiarity plays a part, but a new, powerful 16-bit program should overcome that, surely.

I will continue to work with the Ant, while playing with the Grasshopper, waiting hopefully for it to get down to some real work. With 128s available second-hand for £125 (my *Cass Hyatt*) and lovely printers like the *Panasonic 1880* going for near little quid as £150 (new with cable), it seems that the Ant has a lot of hard work to do yet. New programs and writers are pouring in from America, and I gather that the new 1281 5 1/4-inch disk drive is fast and reliable (as much as the speed and how or five times the capacity of the old 1541 as I've told). Most of the programs available are tried and tested. Spreadsheets, databases, wordprocessors and the like abound, all at prices far lower than similar material for the Amiga and PCs.

So don't sell the C64 or 128 to buy an Amiga. Save up a little longer, and buy an Amiga as well, and while you are waiting continue the good work with the 64 or 128.





# First Steps

*The C128, C16 and Plus4 all have their own sets of strange error messages*

*By Norman Doyle*

**T**he extended limits of the newer Commodore computers mean that extra error messages are required which don't feature on the C64. The C16 Plus4 machines added another one, and that was built on with the advent of the C128 to a further five.

## Can't Resume

*Commodore programming error*

The TRAP statement can be a GOSUB command, but can globally be an interrupt routine. Whenever any error occurs, the program jumps to the error handling subroutine named at whatever line number follows TRAP. The subroutine is terminated by RESUME, which can be a RETURN or a GOSUB routine. If a program is written with the error-handling routine at the end and no END command is encountered, execution will run into the error routine, and when RESUME is encountered the error message will be generated.

## Loop Not Found

*Commodore programming error*

This means that a DO command has been encountered, but the program cannot find a corresponding LOOP command.

## Loop Without Do

*Commodore programming error*

This is the opposite of the previous error. It occurs when a LOOP command is encountered but a DO command isn't currently active.

## Direct Mode Only

*Commodore user error*

Some of the extended Basic commands, such as AUTO, can only

be used in direct mode. In other words, this must be typed onto the screen, and then the RETURN key must be pressed. Any attempt to include such a command in a program will be met with this message.

## No Graphics Area

*Commodore programming error*

This means that the HEADER command was used without an ID to name a disk, without going through the lengthy procedure of formatting. Unfortunately, the disk chains hasn't yet been formatted, so the solution is to repeat the HEADER command with an ID added.

## Bad Disk

*Commodore computer or bad disk*

At best this means that the HEADER command was used with out an ID to name a disk without going through the lengthy procedure of formatting. Unfortunately the disk chains hasn't yet been formatted, so the solution is to repeat the HEADER command with an ID added.

At worst, it means what it says: the disk is faulty. This can be caused through rough or careless handling, but occasionally it means that the disk was faulty in manufacture or simply wore out through over-use. Throw the disk away and start again with a fresh one.

The next group of errors are the ones which only appear on the C128.

## Bend Not Found

*Commodore programming error*

When an IF THEN BEGIN or IF THEN ELSE BEGIN construct is found, the computer then looks for

the corresponding BEND command. If it can't find one, it flags this error.

## Line Too Large

*Commodore user*

The highest possible value for a program line is 65535. If an attempt is made to RENUMBER a program and any of the renumbered line values will exceed 65535, the command is aborted without being executed, and this error message is displayed.

## Unresolved Reference

*Commodore programming error*

If an attempt is made to renumber a program and the programmer has used a line which is referenced by a GOTO GOSUB, ON or TRAP command, the RENUMBER routine will not be able to execute properly. Once again the command is aborted without any action being performed, but this time the unresolved reference message is issued.

## Unimplemented Command

*Commodore program error*

Two commands are included in C128 Basic, but have no entries associated with them. These commands are QUIT and OFF. If either is used, the computer politely reminds the user that it is an unimplemented command.

## File Read

*Commodore operational error*

This means that a file or program has failed to load. The reasons could be because the disk has been damaged or because an error occurred when loading the drive data before the program has finished loading.

# Contributions

*So you own a Commodore? So you've  
written some programs? So why haven't you  
sent them to us?*

**Y**ou'll find *Commodore* is always on the look out for new programs, items and tips, articles and even regular series. In fact, if you have something that you think could be of use to other Commodore owners we want to hear about it.

And if you have got something which you think we may be interested in, how do we go about submitting it to us?

Below you will find a list of guidelines that will help us to deal with any item that you send us to us. We don't expect everybody to be the next William Shakespeare but at least do follow these simple rules that it will make our job a lot easier.

1) If possible all material sent to the magazine should be typed or printed out on a computer printer.

2) All text should be double spaced; a four should be a blank line between each line of text. You should also leave a margin of about 10 characters round the text.

3) On the very first page you should put the following:

Name of the author  
Machina 2000 200  
Any other comments you wish to add  
Year 1  
Year 2  
Year 3

4) The title of the program should be clearly stated  
5) The program should be clearly stated

For example, suppose you had submitted an article on Commodore. You should put something like the following at the head of the page:

Interruptions/Interruptions

5) Please make sure that you do not make any additional marks on your text, especially underlining.

6) Try and write in clear concise English, it does not have to be a work of literature but it must be comprehensible.

7) On the bottom of each page you should put the word MORE if there are more pages to the article or END if it is the last page.

8) If possible include a listing of all programs.

9) Under our correspondence set a staple or hold the pages together. Use a paperclip instead.

10) Programs should be included on either disk or tape. Make sure that you SAVE two copies of every program so that we have a better chance of finding them if problems occur.

11) If you wish to be paid for your work, please state the amount and the method of payment.

12) If you wish to be paid for your work, please state the amount and the method of payment.

13) Photographs if necessary must be either black and white prints or colour slides. We can take them out when we don't worry about this too much.

14) Subscribers of any length are welcome. If you have a free line service that you think may be of use to someone else we welcome it just as much as a full blown one page article.

15) Payment varies quite a lot and depends on quite a number of factors, such as complexity of program, presentation of program, number of magazine pages it takes up etc. Payment is generally between £10.00 and £200.00.

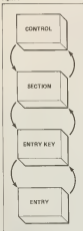
16) All payments are made in the month that the magazine containing your article has appeared in print.

17) If we do find your submission suitable for inclusion in the magazine we will write to you giving the terms of publication, the rate of payment and an agreement form. Please return of this form will allow us to use your program in any or possible.

18) If you wish to be paid for your work, please state the amount and the method of payment.

19) If you wish to be paid for your work, please state the amount and the method of payment.

Figure 1



**E**lectronic Notebook is an electronic implementation of the very popular Personal Organizer. It provides the means by which information can be stored, under a key, in a file. It does this by utilizing database techniques similar to those on minicomputer computers.

An Electronic Notebook file can be as small as or large as you require up to the maximum size of the disk. On the 1541 disk drive this will be around 1500 records, but on larger disks with a capacity of 1 megabyte the number increases to 10000. A single Electronic Notebook file contains four different record types which are used to implement the database structure (See Figure 1). The records form a hierarchical structure in which each level holds varying degrees of information requested by the user. The four records are:

# Electronic Notebook

*Get yourself organised with this powerful database*

*By L. Keighley*

**CONTROL** This record holds control information about the usage of the database. This includes details of when the first empty record is and what the access password is. The user need give access to this record as it is maintained by the Electronic Notebook program alone.

**SECTION** This record is the Header or Folder under which information is to be stored. Again from pointers to the next and last Section records and pointers to its next and last Entry Key records, it holds the Title of the section and its type. The types allowed are Labels, Notes and Text.

**ENTRY KEY** This record holds the key to the information that is to be stored below it, along with the pointer to the last Entry record. Pointers are also held to the next and last Entry Key records.

**ENTRY** Each Entry record holds up to 35 characters of information along with the pointer to the next Entry. If the information to be held is of the Text type a seventy character line is made up of two consecutive Entry records.

The Section records can only be accessed by stepping through all those previous, but the Entry Key records can be obtained using the given Key.

All the records at each level are accessed by a record at the level above. For example a Section record may be designated as 'Addresses' and may have below it two address Entry Keys for 'Fred' and 'Jane' below each of the Entry Key records will be any number of Entry records that it takes to hold all of the address for each person. The records held at all levels below Control are held in 'chains'.

The program allows maintenance of all user information, add, delete, insert at the lowest three levels with the exception of key information this is restricted by password. All the 'chains' are held in sequence of key and to change the key requires that this position in the chain is stored. While the physical position on the disk of the record does not change all the pointers around its old and new positions must be amended. This again requires the reading and writing of a number of records and therefore takes some time.

The database could be used as a diary if each Entry Key within a Section called 'Diary' was given the date of each day in the year.

The hardware output facilities provided by Electronic Notebook are: Labels printed in single or two up form, Notes again printed in single or two up form and Text printed in seventy character lines.

Access to the facilities of Electronic Notebook are from an on-screen menu. Functions are selected by keying the appropriate number. Some require two numbers to be keyed, the first to select general functions and the second to select the specific function.

**Control:**

- E First available empty record
- End of file + 1 record
- S1 First Section record
- S2 Last Section record
- M Maximum number of records
- M1 Master Types used
- PS Password

**Section:**

- T3 Title
- S3 Next Section
- S4 Last Section
- E3 Next Entry
- E4 Last Entry
- T15 Type
- M1 Master

**Entry Key:**

- E4 Next Entry
- S3 Last Entry
- E3 Key
- M1 Next Entry

**Entry:**

- S2 Next Entry
- T3 Data

**Operating Instructions**

- 1 Load and Run the program. **ENTER0000** - a menu will then be displayed of all the screens that are allowed on the database.
- 2 To select an action, with the exception of "Intuitive", requires that two keys be pressed. The first selects the area of the database that is to be worked on or the print option and second selects the type of action.
- 3 When entering Text information the "1" cursor is used. This is because of the way in which the computer takes data in from the keyboard. The cursor is used to indicate the end of input for one field and the beginning of input for the next. So that, the contents can be entered in just the "1" should be used. This will result in a message

being displayed on the screen and printed when necessary.

- 4 If next above, be selected from the menu to end the program that ensures that the database is closed correctly and the last buffer of information is stored on the disk.
- 5 When a new database is set up the number of database records allowed can be varied from 500 up to 1000. Each record has to be written to the database and therefore this takes some time. The database already set up has 1000 records and took approximately 10 minutes to create.
- 6 The password when correctly entered allows the pointer fields to be entered. This can corrupt the database if done incorrectly, hence the password. The password for the sample database is "C64/C64".

## C16 and PLUS '4 PROGRAM

# Head for Home

*Challenge your friends to  
a version of one of the  
World's oldest boardgames*

**H**ead for Home is presented in the form of two Basic loaders. These should be typed in and saved separately if using tape. Make sure that you enter the **LOAD** at the end of the last program to a **LOAD** instruction.

Care should be taken when entering the program as some of the lines contain machine code. A single mistake on entry would cause the program not to work.

**Getting Going**

Once the program is up and running, you will be asked if you require instructions and the number of players. Head for Home is designed for 2-4 players.

To start the game, a player needs to throw a six on the computer dice. The player number around the players will show scores a win. If no move is possible a round will roll you. Once a player throws a six the dice is rolled again, until a different number is thrown. The sum of the dice throws becomes the number of positions that a player can move, though it can't be less than 1 move. A move into the board.

When a move is required a number will appear on the screen. The Player

should press one of the numbers 1

4 depending on which point they wish to move. The move is only accepted if it is legal. If no move can be made press 0 to pass.

Points move in the same order as the die throws. So, if you have two ones and a three and press 3 4 1, pass 2 and 4 move 6 spaces and press 1 moves 4. Entering 2 3 2 would move pass 3 15 spaces.

Each playing piece travels around the board once. When it leaves the white track, and moves onto the last 5 coloured squares. The same number must be thrown in order to reach home base.

That's all there is to it - have fun!

# Software for Sale

*If you think that one of our programs looks very interesting, but you can't afford the time to type it in, then our software service will help you out*

**L**ate three o'clock in the morning. You sit at the computer keyboard having just finished a marathon typing session on one of the superb programs in our *Time Computers* 5, 10 or 15 magazines for the keyboard and print the letters R, T, and N. You press RETURN and nothing happens. Inevitably you probably faced this problem. When it does happen it's a matter of spending hours searching through the program for any typing mistakes. No matter how long you look or how many people help you, you can usually guarantee that you have lost hope deep through a manual.

The Time Computers helpware service makes available all of the programs from each magazine but haven't analysed at a price of £4.00 for disk and £4.00 for manual. None of the documentation for the programs is supplied with the software since it is all available in the relevant magazine. Should you not have the magazine then book means are available from the following address:

**INDINET LTD** 4 River Park House, Berkhamswood, Hertfordshire, Herts.  
Herts. SG11 1HL  
Tel: (04427) 36960

Please contact this address for prices and availability.

## The Disk

Programs on the disk will also be brought up on the working software. As, when possible, we will not use Basic loaders that make one of the programs much easier. Unfortunately at the moment we cannot duplicate C16 and Macintosh disks. However programs for these machines will be available on the disk.

What programs are available?

At the top of each article you will find an explanation of the article type: C16 Program etc. So that you can see which programs are available, as which format, you will also find a couple of symbols after the title. The symbols have the following meaning:



This symbol means that the program is available on cassette



This program is available on disk

## Please Note

None of the programs supplied on cassette, are total working versions of the programs. For the best performance, programs are tape. There is no room in placing a program the is required to be loaded from disk, via no tape.

## OCTOBER 1988

**SPRITE LIBRARY** - In this installment our sprites take on the look of the alphabet (C64)

**SAMPLER 64** - See September 1988 for details (full program on both September and October disks and a tape)

**SET THE ALARM** - Use the C64's on-board clock as an alarm

**JACK IN THE BOX** - A handy box utility for Basic and machine code programmers (C64)

**FILE EXTENSION** - Modify your disk directory so that you can load all your programs with the C64 disk only

**HEURES/MULTICOLOUR PLOTTER** - A solution of plot software for both Basic and machine code programmers (C64)

**JOYSTICK CURSOR** - Let your joystick emulate the cursor keys (C64)

**HE-RES FILE** - A superb machine code file routine for all graphics programmers (C64)

**CODE RELOCATION** - Change your machine code programs so that they will run at any area of your C64's memory

**ORDER CODE**

**TAPE YNOCOTR 0400**

**DISK YNOCOTR 0400**

## NOVEMBER 1988

**SECRET WRITING** - Learn how to conceal messages and how to protect your Basic programs (C64)

**PAINLESS WINDOW'S** - Emulated the power of the C128's WINDOW command

**UNLOCK HIDDEN 100 POWERS** - Use the full layout of your C128 in a C64 mode

**WILLIAM TELL** - Fight off the Austrian soldiers in this great C64 arcade game

**TALKING COMPUTERS** - Use speech synthesizers with your C64

**ORDER CODE**

**TAPE YNOCOTR 0400**

**DISK YNOCOTR 0400**

DECEMBER 1988

**FREE LABELER** Use your Commodore printer to produce a template for your function keys (C64)

**4-D DIRECTORY EDITOR** Rearrange your Plus 4 disk directory entries however you like. Available on disk only  
**DOUBLE HEIGHT** Enhance your C64 text displays with a double height character set.

**PROGRAM PROTECTION** - Protect your latest C64 manuscripts from prying eyes.

**SPRITE CONTROLLER** Basic sprite manipulation made easy (C64)

**GRAPHIC EDITOR** A quick and simple C64 UDG editor.

**MASTERING THE RASTER** The master guide to manipulating your article on raster interrupts (C64)

**RELATING CHARACTERS** - Give your Plus 4 and Commodore printer a better print quality. Available on disk only.

**WILLIAM TELL** See November for details (C64)

**ORDER CODE**

**TAPE YC DEC0814M**

**DISK YD DEC0814M**

JANUARY 1989

**PRIGIAN SPRITES** A powerful sprite editor for the C64  
**DAEPRD** - A simple but helpful text processor for the C64. Available on disk and cassette but will only store files on tape.

**UDG COMPRESSOR** Save on memory when using UDG's in your programs. For C64 only.

**WILLIAM TELL** Our popular arcade game for the C64  
**4-D ACTORUN** Improve tape loading on your Plus 4 cassette. Only available on disk.  
**MINIBASE** A database for C128 owners.

**ORDER CODE**

**DISK YD JAN0914M**

**TAPE YC JAN0914M**

FEBRUARY 1989

**TAPE MENU** Add a menu system to your program cassette (C64)

**SONIC EFFECTS** A superb sound editor for the C64  
**P BUMP** Dump your C64 text screens to printer with ease.

**DATA LOADER** A simple way to load those ranges of C64 DATA files.

**SPRITE LIBRARY** A collection of hands to your growing library (C64)

**PLAY THE GAME** A superb first machine program for the Plus 4. (Available on disk only)

**ORDER CODE**

**DISK YD FEB0914M**

**TAPE YC FEB0914M**

Cassette or disks are available from March 1989. Please see the editorial of issue 181 317 31264 for details of files.

ORDER FORM - PLEASE COMPLETE IN BLOCK CAPITALS

NAME	QTY	TAPE/DISK	ORDER CODE	PRICE
MARCH 89		TAPE (£4.00)	YCMAR89	
MARCH 89		DISK (£6.00)	YDMAR89	
OVERSEAS POST £1				
			TOTAL	

NAME

ADDRESS

POSTCODE

I enclose a cheque/postal order for £  made payable to ARGUS SPECIALIST PUBLICATIONS LTD.

All orders should be sent to: YOUR COMMODORE READERS SERVICES, ARGUS SPECIALIST PUBLICATIONS, 9 HALL ROAD, HEMEL HEMPSTEAD, HERTS HP8 7QH

Please allow 28 days for delivery.



## Checksum Program

The hexadecimal numbers appearing in a column to the left of the listings should not be typed in with the program. These are merely checksum values and are there to help you get each line right. Don't worry if you don't understand the hexadecimal system as long as you can compare two characters on the screen with the corresponding two characters in the magazine; you can use our line-checking program.

Type in the Checksum Program, make sure that you've not made any mistakes and save it in tape or disk

immediately, because it will be used with most of the programs and listings appearing in *Your Commodore*.

At the start of each programming session, load Checksum and run it. The screen will first show two uppercase characters and each time you type in a line, and press the RETURN key, a number will appear on the screen in white. This should be identical to the corresponding value in the magazine.

















If the two values don't coincide in one another, you have not typed the line exactly as printed so go back and check each character carefully. When you find the error simply correct it and

press RETURN again.

If you want to run with double-height type, type `SHS49152` and the screen will be tinted in the familiar blue colours. You can then do whatever it was you wanted to do and if this doesn't work the area where Checksum lives you can go back to it with the same `SHS` command.

No system is foolproof but the chances of two errors cancelling one another out are pretty small. Most of the listings are presented in lower case. To save your computer to lower case mode, press the Commodore key and the SHIFT key at the same time.

### Mnemonic Symbol Keypress

[RIGHT]		CTRL left/right
[LEFT]		SHIFT & CTRL left/right
[DOWN]		CTRL up/down
[UP]		SHIFT & CTRL up/down
[F1]		F1 key
[F2]		SHIFT & F1 key
[F3]		F3 key
[F4]		SHIFT & F3 key
[F5]		F5 key
[F6]		SHIFT & F5 key
[F7]		F7 key
[F8]		SHIFT & F7 key
[HOME]		CLR/HOME
[CLR]		SHIFT & CLR/HOME
[RYSON]		CTRL & 9
[RYSOFF]		CTRL & 0

### Mnemonic Symbol Keypress

[BASIC]		CTRL & 1
[WRITE]		CTRL & 2
[RED]		CTRL & 3
[CYAN]		CTRL & 4
[PURPLE]		CTRL & 5
[GREEN]		CTRL & 6
[MAG]		CTRL & 7
[YELLOW]		CTRL & 8
[POUND]		1
[LARGROW]		4
[UPARROW]		↑
[P8]		SHIFT & 8
[INVT]		SHIFT & INVT/DEL
[REV T]		rev text
[TANAL]		CRM + tanal
[NANAL]		SHIFT + tanal

LIFESAVER	COM	REVERSER	101
<p>This 8086 machine code routine will enable Basic programs to be enhanced with flashing messages etc. It will reserve on and off any part of the screen that you desire.</p> <p>The routine redirects the IRQ interrupt Vector of memory location 788 to your Basic program will be free to do what you want.</p> <p>To start REVERSER, POKE the data into memory then POKE the following locations with the values requested:            POKE 151,1-BYTE (Screen start address)            POKE 152,16-BYTE            POKE 153,NUMBER OF CHARS            POKE 154,FLASH RATE            Then SYS 49132 to start or            SYS 49150 to switch off.</p> <p>The program includes a short demo to show how to set the program up.</p>			
<pre> 00000000 **** REVERSER MACHINE CODE **** 20 F0H&lt;49150H&gt;151 0000A F0H&lt;A 0004A M00T 70 B0D&lt;0007H&gt;0000H&lt;DATA ERROR&gt; STOP 40 DATA 170 140 13 141 20 1 000 000 141 21 1 0000 170 07 000 50 DATA 170 154 000 18 140 200 141 37 150 160 0 170 151 71 120 140 60 DATA 211 200 190 151 000 140 210 21 100 70 40 190 130 160 40 140 70 DATA 20 1 160 170 141 21 1 000 00 1 160 150 0 1 11 70 80 REVERSER DEMO **** 90 PORTF0H&lt;0 F0H&gt;1510 F0H 140 1 100 PRINT CHR\$(4); PRINT TAB(17);"0000 COMANDER" 110 F0H151 1 F0H 1 NUMBER OF CHARACTERS TO REVERSE 1 120 F0H152 1 F0H 1 FLASH RATE 1 130 A=1510 F0H 1 SCREEN ADDRESS 1 140 F0H151 AND 150 F0H 1 F0H 1 VALUE FOR SCREEN ADDRESS 1 150 F0H151 1 F0H 160 SYS 49132 F0H 1 CALL MC 1 170 GET AS 0000 1 F0H 170 180 SYS 49150 F0H 1 SWITCH OFF MC 1 </pre>			
By Neil Higgins			

# Listings

## William Tell



### PROGRAM WILLIAM TELL LISTING 3

```

1 10 REMARKS 10000 000000
2
3 40 FOR I=0 TO 10: FOR J=0 TO 10:
4 50 NEXT J
5 60 PRINT "PROGRAMMING A"
6 70 IF CY=0 THEN POKE 151,100:GOTO 1
7 80
8 90 FOR I=0 TO 10: FOR J=0 TO 10:
9 100 NEXT J
10 110 PRINT "STOP"
11 120
12 130 PRINT "PROGRAMMING A"
13 140 PRINT "STOP"
14 150 PRINT "PROGRAMMING A"
15 160 PRINT "STOP"
16 170 PRINT "PROGRAMMING A"
17 180 PRINT "STOP"
18 190 PRINT "PROGRAMMING A"
19 200 PRINT "STOP"
20 210 PRINT "PROGRAMMING A"
21 220 PRINT "STOP"
22 230 PRINT "PROGRAMMING A"
23 240 PRINT "STOP"
24 250 PRINT "PROGRAMMING A"
25 260 PRINT "STOP"
26 270 PRINT "PROGRAMMING A"
27 280 PRINT "STOP"
28 290 PRINT "PROGRAMMING A"
29 300 PRINT "STOP"
30 310 PRINT "PROGRAMMING A"
31 320 PRINT "STOP"
32 330 PRINT "PROGRAMMING A"
33 340 PRINT "STOP"
34 350 PRINT "PROGRAMMING A"
35 360 PRINT "STOP"
36 370 PRINT "PROGRAMMING A"
37 380 PRINT "STOP"
38 390 PRINT "PROGRAMMING A"
39 400 PRINT "STOP"
40 410 PRINT "PROGRAMMING A"
41 420 PRINT "STOP"
42 430 PRINT "PROGRAMMING A"
43 440 PRINT "STOP"
44 450 PRINT "PROGRAMMING A"
45 460 PRINT "STOP"
46 470 PRINT "PROGRAMMING A"
47 480 PRINT "STOP"
48 490 PRINT "PROGRAMMING A"
49 500 PRINT "STOP"
50 510 PRINT "PROGRAMMING A"
51 520 PRINT "STOP"
52 530 PRINT "PROGRAMMING A"
53 540 PRINT "STOP"
54 550 PRINT "PROGRAMMING A"
55 560 PRINT "STOP"
56 570 PRINT "PROGRAMMING A"
57 580 PRINT "STOP"
58 590 PRINT "PROGRAMMING A"
59 600 PRINT "STOP"
60 610 PRINT "PROGRAMMING A"
61 620 PRINT "STOP"
62 630 PRINT "PROGRAMMING A"
63 640 PRINT "STOP"
64 650 PRINT "PROGRAMMING A"
65 660 PRINT "STOP"
66 670 PRINT "PROGRAMMING A"
67 680 PRINT "STOP"
68 690 PRINT "PROGRAMMING A"
69 700 PRINT "STOP"
70 710 PRINT "PROGRAMMING A"
71 720 PRINT "STOP"
72 730 PRINT "PROGRAMMING A"
73 740 PRINT "STOP"
74 750 PRINT "PROGRAMMING A"
75 760 PRINT "STOP"
76 770 PRINT "PROGRAMMING A"
77 780 PRINT "STOP"
78 790 PRINT "PROGRAMMING A"
79 800 PRINT "STOP"
80 810 PRINT "PROGRAMMING A"
81 820 PRINT "STOP"
82 830 PRINT "PROGRAMMING A"
83 840 PRINT "STOP"
84 850 PRINT "PROGRAMMING A"
85 860 PRINT "STOP"
86 870 PRINT "PROGRAMMING A"
87 880 PRINT "STOP"
88 890 PRINT "PROGRAMMING A"
89 900 PRINT "STOP"
90 910 PRINT "PROGRAMMING A"
91 920 PRINT "STOP"
92 930 PRINT "PROGRAMMING A"
93 940 PRINT "STOP"
94 950 PRINT "PROGRAMMING A"
95 960 PRINT "STOP"
96 970 PRINT "PROGRAMMING A"
97 980 PRINT "STOP"
98 990 PRINT "PROGRAMMING A"
99 1000 PRINT "STOP"

```

```

1 10 REMARKS 10000 000000
2
3 40 FOR I=0 TO 10: FOR J=0 TO 10:
4 50 NEXT J
5 60 PRINT "PROGRAMMING A"
6 70 IF CY=0 THEN POKE 151,100:GOTO 1
7 80
8 90 FOR I=0 TO 10: FOR J=0 TO 10:
9 100 NEXT J
10 110 PRINT "STOP"
11 120
12 130 PRINT "PROGRAMMING A"
13 140 PRINT "STOP"
14 150 PRINT "PROGRAMMING A"
15 160 PRINT "STOP"
16 170 PRINT "PROGRAMMING A"
17 180 PRINT "STOP"
18 190 PRINT "PROGRAMMING A"
19 200 PRINT "STOP"
20 210 PRINT "PROGRAMMING A"
21 220 PRINT "STOP"
22 230 PRINT "PROGRAMMING A"
23 240 PRINT "STOP"
24 250 PRINT "PROGRAMMING A"
25 260 PRINT "STOP"
26 270 PRINT "PROGRAMMING A"
27 280 PRINT "STOP"
28 290 PRINT "PROGRAMMING A"
29 300 PRINT "STOP"
30 310 PRINT "PROGRAMMING A"
31 320 PRINT "STOP"
32 330 PRINT "PROGRAMMING A"
33 340 PRINT "STOP"
34 350 PRINT "PROGRAMMING A"
35 360 PRINT "STOP"
36 370 PRINT "PROGRAMMING A"
37 380 PRINT "STOP"
38 390 PRINT "PROGRAMMING A"
39 400 PRINT "STOP"
40 410 PRINT "PROGRAMMING A"
41 420 PRINT "STOP"
42 430 PRINT "PROGRAMMING A"
43 440 PRINT "STOP"
44 450 PRINT "PROGRAMMING A"
45 460 PRINT "STOP"
46 470 PRINT "PROGRAMMING A"
47 480 PRINT "STOP"
48 490 PRINT "PROGRAMMING A"
49 500 PRINT "STOP"
50 510 PRINT "PROGRAMMING A"
51 520 PRINT "STOP"
52 530 PRINT "PROGRAMMING A"
53 540 PRINT "STOP"
54 550 PRINT "PROGRAMMING A"
55 560 PRINT "STOP"
56 570 PRINT "PROGRAMMING A"
57 580 PRINT "STOP"
58 590 PRINT "PROGRAMMING A"
59 600 PRINT "STOP"
60 610 PRINT "PROGRAMMING A"
61 620 PRINT "STOP"
62 630 PRINT "PROGRAMMING A"
63 640 PRINT "STOP"
64 650 PRINT "PROGRAMMING A"
65 660 PRINT "STOP"
66 670 PRINT "PROGRAMMING A"
67 680 PRINT "STOP"
68 690 PRINT "PROGRAMMING A"
69 700 PRINT "STOP"
70 710 PRINT "PROGRAMMING A"
71 720 PRINT "STOP"
72 730 PRINT "PROGRAMMING A"
73 740 PRINT "STOP"
74 750 PRINT "PROGRAMMING A"
75 760 PRINT "STOP"
76 770 PRINT "PROGRAMMING A"
77 780 PRINT "STOP"
78 790 PRINT "PROGRAMMING A"
79 800 PRINT "STOP"
80 810 PRINT "PROGRAMMING A"
81 820 PRINT "STOP"
82 830 PRINT "PROGRAMMING A"
83 840 PRINT "STOP"
84 850 PRINT "PROGRAMMING A"
85 860 PRINT "STOP"
86 870 PRINT "PROGRAMMING A"
87 880 PRINT "STOP"
88 890 PRINT "PROGRAMMING A"
89 900 PRINT "STOP"
90 910 PRINT "PROGRAMMING A"
91 920 PRINT "STOP"
92 930 PRINT "PROGRAMMING A"
93 940 PRINT "STOP"
94 950 PRINT "PROGRAMMING A"
95 960 PRINT "STOP"
96 970 PRINT "PROGRAMMING A"
97 980 PRINT "STOP"
98 990 PRINT "PROGRAMMING A"
99 1000 PRINT "STOP"

```

```

1 10 REMARKS 10000 000000
2
3 40 FOR I=0 TO 10: FOR J=0 TO 10:
4 50 NEXT J
5 60 PRINT "PROGRAMMING A"
6 70 IF CY=0 THEN POKE 151,100:GOTO 1
7 80
8 90 FOR I=0 TO 10: FOR J=0 TO 10:
9 100 NEXT J
10 110 PRINT "STOP"
11 120
12 130 PRINT "PROGRAMMING A"
13 140 PRINT "STOP"
14 150 PRINT "PROGRAMMING A"
15 160 PRINT "STOP"
16 170 PRINT "PROGRAMMING A"
17 180 PRINT "STOP"
18 190 PRINT "PROGRAMMING A"
19 200 PRINT "STOP"
20 210 PRINT "PROGRAMMING A"
21 220 PRINT "STOP"
22 230 PRINT "PROGRAMMING A"
23 240 PRINT "STOP"
24 250 PRINT "PROGRAMMING A"
25 260 PRINT "STOP"
26 270 PRINT "PROGRAMMING A"
27 280 PRINT "STOP"
28 290 PRINT "PROGRAMMING A"
29 300 PRINT "STOP"
30 310 PRINT "PROGRAMMING A"
31 320 PRINT "STOP"
32 330 PRINT "PROGRAMMING A"
33 340 PRINT "STOP"
34 350 PRINT "PROGRAMMING A"
35 360 PRINT "STOP"
36 370 PRINT "PROGRAMMING A"
37 380 PRINT "STOP"
38 390 PRINT "PROGRAMMING A"
39 400 PRINT "STOP"
40 410 PRINT "PROGRAMMING A"
41 420 PRINT "STOP"
42 430 PRINT "PROGRAMMING A"
43 440 PRINT "STOP"
44 450 PRINT "PROGRAMMING A"
45 460 PRINT "STOP"
46 470 PRINT "PROGRAMMING A"
47 480 PRINT "STOP"
48 490 PRINT "PROGRAMMING A"
49 500 PRINT "STOP"
50 510 PRINT "PROGRAMMING A"
51 520 PRINT "STOP"
52 530 PRINT "PROGRAMMING A"
53 540 PRINT "STOP"
54 550 PRINT "PROGRAMMING A"
55 560 PRINT "STOP"
56 570 PRINT "PROGRAMMING A"
57 580 PRINT "STOP"
58 590 PRINT "PROGRAMMING A"
59 600 PRINT "STOP"
60 610 PRINT "PROGRAMMING A"
61 620 PRINT "STOP"
62 630 PRINT "PROGRAMMING A"
63 640 PRINT "STOP"
64 650 PRINT "PROGRAMMING A"
65 660 PRINT "STOP"
66 670 PRINT "PROGRAMMING A"
67 680 PRINT "STOP"
68 690 PRINT "PROGRAMMING A"
69 700 PRINT "STOP"
70 710 PRINT "PROGRAMMING A"
71 720 PRINT "STOP"
72 730 PRINT "PROGRAMMING A"
73 740 PRINT "STOP"
74 750 PRINT "PROGRAMMING A"
75 760 PRINT "STOP"
76 770 PRINT "PROGRAMMING A"
77 780 PRINT "STOP"
78 790 PRINT "PROGRAMMING A"
79 800 PRINT "STOP"
80 810 PRINT "PROGRAMMING A"
81 820 PRINT "STOP"
82 830 PRINT "PROGRAMMING A"
83 840 PRINT "STOP"
84 850 PRINT "PROGRAMMING A"
85 860 PRINT "STOP"
86 870 PRINT "PROGRAMMING A"
87 880 PRINT "STOP"
88 890 PRINT "PROGRAMMING A"
89 900 PRINT "STOP"
90 910 PRINT "PROGRAMMING A"
91 920 PRINT "STOP"
92 930 PRINT "PROGRAMMING A"
93 940 PRINT "STOP"
94 950 PRINT "PROGRAMMING A"
95 960 PRINT "STOP"
96 970 PRINT "PROGRAMMING A"
97 980 PRINT "STOP"
98 990 PRINT "PROGRAMMING A"
99 1000 PRINT "STOP"

```



10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

**Table 1**

100

## LISTINGS

[illegible]

10

YOUR COMMENT: [comment on this story](#)

[illegible][illegible][illegible]





## LISTINGS

[illegible]

# LISTINGS

24	741 12 1782	95	741 12 1782	17	741 12 1782
25	741 12 1782	96	741 12 1782	18	741 12 1782
26	741 12 1782	97	741 12 1782	19	741 12 1782
27	741 12 1782	98	741 12 1782	20	741 12 1782
28	741 12 1782	99	741 12 1782	21	741 12 1782
29	741 12 1782	100	741 12 1782	22	741 12 1782
30	741 12 1782	101	741 12 1782	23	741 12 1782
31	741 12 1782	102	741 12 1782	24	741 12 1782
32	741 12 1782	103	741 12 1782	25	741 12 1782
33	741 12 1782	104	741 12 1782	26	741 12 1782
34	741 12 1782	105	741 12 1782	27	741 12 1782
35	741 12 1782	106	741 12 1782	28	741 12 1782
36	741 12 1782	107	741 12 1782	29	741 12 1782
37	741 12 1782	108	741 12 1782	30	741 12 1782
38	741 12 1782	109	741 12 1782	31	741 12 1782
39	741 12 1782	110	741 12 1782	32	741 12 1782
40	741 12 1782	111	741 12 1782	33	741 12 1782
41	741 12 1782	112	741 12 1782	34	741 12 1782
42	741 12 1782	113	741 12 1782	35	741 12 1782
43	741 12 1782	114	741 12 1782	36	741 12 1782
44	741 12 1782	115	741 12 1782	37	741 12 1782
45	741 12 1782	116	741 12 1782	38	741 12 1782
46	741 12 1782	117	741 12 1782	39	741 12 1782
47	741 12 1782	118	741 12 1782	40	741 12 1782
48	741 12 1782	119	741 12 1782	41	741 12 1782
49	741 12 1782	120	741 12 1782	42	741 12 1782
50	741 12 1782	121	741 12 1782	43	741 12 1782
51	741 12 1782	122	741 12 1782	44	741 12 1782
52	741 12 1782	123	741 12 1782	45	741 12 1782
53	741 12 1782	124	741 12 1782	46	741 12 1782
54	741 12 1782	125	741 12 1782	47	741 12 1782
55	741 12 1782	126	741 12 1782	48	741 12 1782
56	741 12 1782	127	741 12 1782	49	741 12 1782
57	741 12 1782	128	741 12 1782	50	741 12 1782
58	741 12 1782	129	741 12 1782	51	741 12 1782
59	741 12 1782	130	741 12 1782	52	741 12 1782
60	741 12 1782	131	741 12 1782	53	741 12 1782
61	741 12 1782	132	741 12 1782	54	741 12 1782
62	741 12 1782	133	741 12 1782	55	741 12 1782
63	741 12 1782	134	741 12 1782	56	741 12 1782
64	741 12 1782	135	741 12 1782	57	741 12 1782
65	741 12 1782	136	741 12 1782	58	741 12 1782
66	741 12 1782	137	741 12 1782	59	741 12 1782
67	741 12 1782	138	741 12 1782	60	741 12 1782
68	741 12 1782	139	741 12 1782	61	741 12 1782
69	741 12 1782	140	741 12 1782	62	741 12 1782
70	741 12 1782	141	741 12 1782	63	741 12 1782
71	741 12 1782	142	741 12 1782	64	741 12 1782
72	741 12 1782	143	741 12 1782	65	741 12 1782
73	741 12 1782	144	741 12 1782	66	741 12 1782
74	741 12 1782	145	741 12 1782	67	741 12 1782
75	741 12 1782	146	741 12 1782	68	741 12 1782
76	741 12 1782	147	741 12 1782	69	741 12 1782
77	741 12 1782	148	741 12 1782	70	741 12 1782
78	741 12 1782	149	741 12 1782	71	741 12 1782
79	741 12 1782	150	741 12 1782	72	741 12 1782
80	741 12 1782	151	741 12 1782	73	741 12 1782
81	741 12 1782	152	741 12 1782	74	741 12 1782
82	741 12 1782	153	741 12 1782	75	741 12 1782
83	741 12 1782	154	741 12 1782	76	741 12 1782
84	741 12 1782	155	741 12 1782	77	741 12 1782
85	741 12 1782	156	741 12 1782	78	741 12 1782
86	741 12 1782	157	741 12 1782	79	741 12 1782
87	741 12 1782	158	741 12 1782	80	741 12 1782
88	741 12 1782	159	741 12 1782	81	741 12 1782
89	741 12 1782	160	741 12 1782	82	741 12 1782
90	741 12 1782	161	741 12 1782	83	741 12 1782
91	741 12 1782	162	741 12 1782	84	741 12 1782
92	741 12 1782	163	741 12 1782	85	741 12 1782
93	741 12 1782	164	741 12 1782	86	741 12 1782
94	741 12 1782	165	741 12 1782	87	741 12 1782
95	741 12 1782	166	741 12 1782	88	741 12 1782
96	741 12 1782	167	741 12 1782	89	741 12 1782
97	741 12 1782	168	741 12 1782	90	741 12 1782
98	741 12 1782	169	741 12 1782	91	741 12 1782
99	741 12 1782	170	741 12 1782	92	741 12 1782
100	741 12 1782	171	741 12 1782	93	741 12 1782
		172	741 12 1782	94	741 12 1782
		173	741 12 1782	95	741 12 1782
		174	741 12 1782	96	741 12 1782
		175	741 12 1782	97	741 12 1782
		176	741 12 1782	98	741 12 1782
		177	741 12 1782	99	741 12 1782
		178	741 12 1782	100	741 12 1782



[illegible]

DOI: 10.1002/for

[illegible]

## NOTES

```

05 40 PRINT "WIRE TEST COMPLE"
10 40 PRINT "END OF WIRE TEST BE"
    WILL CHANGE THE CONNECTION TO
    THE LAST OF THE CUBION.
20 40 PRINT "IF THERE IS NO
    TEST
30 40 PRINT "INSTEAD THE CUBION
    7 LINE. IT WILL EXIST THE LA
    ST CONNECTION FOR
40 40 PRINT "THE LINE ABOVE (DO
    NOT)
50 40 PRINT "RECONNECTING THE AL-
    LIGMENT NOW (UP TO 1000)
    DO 40 DO 40 END: REPEAT THE C
    USSION
60 40 PRINT "TO THE END OF THE
    TEST LINE ABOVE. RETURN

```

[illegible][illegible]

```

41 01 1000 11-10 1-01 0101-010101
42 011-01 1-10-10-00
43 01 010101
44 02 01 1-01 1-01 01-01 01-01 101 01
45 02 011 1-01
46 02 010101010101 100101 01 01
47 01 101 01-01 01 011 01-01
48 01 0101 1011 101 10101011
49 01 0101 1011 10101011 1011 10101011
50 01 0101 10101011 10101011 10101011
51 01 0101 10101011 10101011 10101011
52 01 0101 10101011 10101011 10101011
53 01 0101 10101011 10101011 10101011
54 01 0101 10101011 10101011 10101011
55 01 0101 10101011 10101011 10101011
56 01 0101 10101011 10101011 10101011
57 01 0101 10101011 10101011 10101011
58 01 0101 10101011 10101011 10101011
59 01 0101 10101011 10101011 10101011
60 01 0101 10101011 10101011 10101011
61 01 0101 10101011 10101011 10101011
62 01 0101 10101011 10101011 10101011
63 01 0101 10101011 10101011 10101011
64 01 0101 10101011 10101011 10101011
65 01 0101 10101011 10101011 10101011
66 01 0101 10101011 10101011 10101011
67 01 0101 10101011 10101011 10101011
68 01 0101 10101011 10101011 10101011
69 01 0101 10101011 10101011 10101011
70 01 0101 10101011 10101011 10101011
71 01 0101 10101011 10101011 10101011
72 01 0101 10101011 10101011 10101011
73 01 0101 10101011 10101011 10101011
74 01 0101 10101011 10101011 10101011
75 01 0101 10101011 10101011 10101011
76 01 0101 10101011 10101011 10101011
77 01 0101 10101011 10101011 10101011
78 01 0101 10101011 10101011 10101011
79 01 0101 10101011 10101011 10101011
80 01 0101 10101011 10101011 10101011
81 01 0101 10101011 10101011 10101011
82 01 0101 10101011 10101011 10101011
83 01 0101 10101011 10101011 10101011
84 01 0101 10101011 10101011 10101011
85 01 0101 10101011 10101011 10101011
86 01 0101 10101011 10101011 10101011
87 01 0101 10101011 10101011 10101011
88 01 0101 10101011 10101011 10101011
89 01 0101 10101011 10101011 10101011
90 01 0101 10101011 10101011 10101011
91 01 0101 10101011 10101011 10101011
92 01 0101 10101011 10101011 10101011
93 01 0101 10101011 10101011 10101011
94 01 0101 10101011 10101011 10101011
95 01 0101 10101011 10101011 10101011
96 01 0101 10101011 10101011 10101011
97 01 0101 10101011 10101011 10101011
98 01 0101 10101011 10101011 10101011
99 01 0101 10101011 10101011 10101011
100 01 0101 10101011 10101011 10101011

```









## LISTINGS

[illegible]





## LISTINGS

[illegible]

## LISTINGS

[illegible]

# LISTINGS

67	WFO 1000-1000-1000-1000	67	WFO 1000-1000-1000-1000
68	WFO 1000-1000-1000-1000	68	WFO 1000-1000-1000-1000
69	WFO 1000-1000-1000-1000	69	WFO 1000-1000-1000-1000
70	WFO 1000-1000-1000-1000	70	WFO 1000-1000-1000-1000
71	WFO 1000-1000-1000-1000	71	WFO 1000-1000-1000-1000
72	WFO 1000-1000-1000-1000	72	WFO 1000-1000-1000-1000
73	WFO 1000-1000-1000-1000	73	WFO 1000-1000-1000-1000
74	WFO 1000-1000-1000-1000	74	WFO 1000-1000-1000-1000
75	WFO 1000-1000-1000-1000	75	WFO 1000-1000-1000-1000
76	WFO 1000-1000-1000-1000	76	WFO 1000-1000-1000-1000
77	WFO 1000-1000-1000-1000	77	WFO 1000-1000-1000-1000
78	WFO 1000-1000-1000-1000	78	WFO 1000-1000-1000-1000
79	WFO 1000-1000-1000-1000	79	WFO 1000-1000-1000-1000
80	WFO 1000-1000-1000-1000	80	WFO 1000-1000-1000-1000
81	WFO 1000-1000-1000-1000	81	WFO 1000-1000-1000-1000
82	WFO 1000-1000-1000-1000	82	WFO 1000-1000-1000-1000
83	WFO 1000-1000-1000-1000	83	WFO 1000-1000-1000-1000
84	WFO 1000-1000-1000-1000	84	WFO 1000-1000-1000-1000
85	WFO 1000-1000-1000-1000	85	WFO 1000-1000-1000-1000
86	WFO 1000-1000-1000-1000	86	WFO 1000-1000-1000-1000
87	WFO 1000-1000-1000-1000	87	WFO 1000-1000-1000-1000
88	WFO 1000-1000-1000-1000	88	WFO 1000-1000-1000-1000
89	WFO 1000-1000-1000-1000	89	WFO 1000-1000-1000-1000
90	WFO 1000-1000-1000-1000	90	WFO 1000-1000-1000-1000
91	WFO 1000-1000-1000-1000	91	WFO 1000-1000-1000-1000
92	WFO 1000-1000-1000-1000	92	WFO 1000-1000-1000-1000
93	WFO 1000-1000-1000-1000	93	WFO 1000-1000-1000-1000
94	WFO 1000-1000-1000-1000	94	WFO 1000-1000-1000-1000
95	WFO 1000-1000-1000-1000	95	WFO 1000-1000-1000-1000
96	WFO 1000-1000-1000-1000	96	WFO 1000-1000-1000-1000
97	WFO 1000-1000-1000-1000	97	WFO 1000-1000-1000-1000
98	WFO 1000-1000-1000-1000	98	WFO 1000-1000-1000-1000
99	WFO 1000-1000-1000-1000	99	WFO 1000-1000-1000-1000
100	WFO 1000-1000-1000-1000	100	WFO 1000-1000-1000-1000

WITTE CONSUMPTION: 1990-1991

## LISTINGS

[illegible]

```

78 ROM GETIN (PUSH) * ROMGET
79 ROM GETIN (PUSH) * ROMGET
80 ROM GETIN (PUSH) * ROMGET
81 ROM GETIN (PUSH) * ROMGET
82 ROM GETIN (PUSH) * ROMGET
83 ROM GETIN (PUSH) * ROMGET
84 ROM GETIN (PUSH) * ROMGET
85 ROM GETIN (PUSH) * ROMGET
86 ROM GETIN (PUSH) * ROMGET
87 ROM GETIN (PUSH) * ROMGET
88 ROM GETIN (PUSH) * ROMGET
89 ROM GETIN (PUSH) * ROMGET
90 ROM GETIN (PUSH) * ROMGET
91 ROM GETIN (PUSH) * ROMGET
92 ROM GETIN (PUSH) * ROMGET
93 ROM GETIN (PUSH) * ROMGET
94 ROM GETIN (PUSH) * ROMGET
95 ROM GETIN (PUSH) * ROMGET
96 ROM GETIN (PUSH) * ROMGET
97 ROM GETIN (PUSH) * ROMGET
98 ROM GETIN (PUSH) * ROMGET
99 ROM GETIN (PUSH) * ROMGET
100 ROM GETIN (PUSH) * ROMGET
101 ROM GETIN (PUSH) * ROMGET
102 ROM GETIN (PUSH) * ROMGET
103 ROM GETIN (PUSH) * ROMGET
104 ROM GETIN (PUSH) * ROMGET
105 ROM GETIN (PUSH) * ROMGET
106 ROM GETIN (PUSH) * ROMGET
107 ROM GETIN (PUSH) * ROMGET
108 ROM GETIN (PUSH) * ROMGET
109 ROM GETIN (PUSH) * ROMGET
110 ROM GETIN (PUSH) * ROMGET
111 ROM GETIN (PUSH) * ROMGET
112 ROM GETIN (PUSH) * ROMGET
113 ROM GETIN (PUSH) * ROMGET
114 ROM GETIN (PUSH) * ROMGET
115 ROM GETIN (PUSH) * ROMGET
116 ROM GETIN (PUSH) * ROMGET
117 ROM GETIN (PUSH) * ROMGET
118 ROM GETIN (PUSH) * ROMGET
119 ROM GETIN (PUSH) * ROMGET
120 ROM GETIN (PUSH) * ROMGET
121 ROM GETIN (PUSH) * ROMGET
122 ROM GETIN (PUSH) * ROMGET
123 ROM GETIN (PUSH) * ROMGET
124 ROM GETIN (PUSH) * ROMGET
125 ROM GETIN (PUSH) * ROMGET
126 ROM GETIN (PUSH) * ROMGET
127 ROM GETIN (PUSH) * ROMGET
128 ROM GETIN (PUSH) * ROMGET
129 ROM GETIN (PUSH) * ROMGET
130 ROM GETIN (PUSH) * ROMGET
131 ROM GETIN (PUSH) * ROMGET
132 ROM GETIN (PUSH) * ROMGET
133 ROM GETIN (PUSH) * ROMGET
134 ROM GETIN (PUSH) * ROMGET
135 ROM GETIN (PUSH) * ROMGET
136 ROM GETIN (PUSH) * ROMGET
137 ROM GETIN (PUSH) * ROMGET
138 ROM GETIN (PUSH) * ROMGET
139 ROM GETIN (PUSH) * ROMGET
140 ROM GETIN (PUSH) * ROMGET
141 ROM GETIN (PUSH) * ROMGET
142 ROM GETIN (PUSH) * ROMGET
143 ROM GETIN (PUSH) * ROMGET
144 ROM GETIN (PUSH) * ROMGET
145 ROM GETIN (PUSH) * ROMGET
146 ROM GETIN (PUSH) * ROMGET
147 ROM GETIN (PUSH) * ROMGET
148 ROM GETIN (PUSH) * ROMGET
149 ROM GETIN (PUSH) * ROMGET
150 ROM GETIN (PUSH) * ROMGET
151 ROM GETIN (PUSH) * ROMGET
152 ROM GETIN (PUSH) * ROMGET
153 ROM GETIN (PUSH) * ROMGET
154 ROM GETIN (PUSH) * ROMGET
155 ROM GETIN (PUSH) * ROMGET
156 ROM GETIN (PUSH) * ROMGET
157 ROM GETIN (PUSH) * ROMGET
158 ROM GETIN (PUSH) * ROMGET
159 ROM GETIN (PUSH) * ROMGET
160 ROM GETIN (PUSH) * ROMGET
161 ROM GETIN (PUSH) * ROMGET
162 ROM GETIN (PUSH) * ROMGET
163 ROM GETIN (PUSH) * ROMGET
164 ROM GETIN (PUSH) * ROMGET
165 ROM GETIN (PUSH) * ROMGET
166 ROM GETIN (PUSH) * ROMGET
167 ROM GETIN (PUSH) * ROMGET
168 ROM GETIN (PUSH) * ROMGET
169 ROM GETIN (PUSH) * ROMGET
170 ROM GETIN (PUSH) * ROMGET
171 ROM GETIN (PUSH) * ROMGET
172 ROM GETIN (PUSH) * ROMGET
173 ROM GETIN (PUSH) * ROMGET
174 ROM GETIN (PUSH) * ROMGET
175 ROM GETIN (PUSH) * ROMGET
176 ROM GETIN (PUSH) * ROMGET
177 ROM GETIN (PUSH) * ROMGET
178 ROM GETIN (PUSH) * ROMGET
179 ROM GETIN (PUSH) * ROMGET
180 ROM GETIN (PUSH) * ROMGET
181 ROM GETIN (PUSH) * ROMGET
182 ROM GETIN (PUSH) * ROMGET
183 ROM GETIN (PUSH) * ROMGET
184 ROM GETIN (PUSH) * ROMGET
185 ROM GETIN (PUSH) * ROMGET
186 ROM GETIN (PUSH) * ROMGET
187 ROM GETIN (PUSH) * ROMGET
188 ROM GETIN (PUSH) * ROMGET
189 ROM GETIN (PUSH) * ROMGET
190 ROM GETIN (PUSH) * ROMGET
191 ROM GETIN (PUSH) * ROMGET
192 ROM GETIN (PUSH) * ROMGET
193 ROM GETIN (PUSH) * ROMGET
194 ROM GETIN (PUSH) * ROMGET
195 ROM GETIN (PUSH) * ROMGET
196 ROM GETIN (PUSH) * ROMGET
197 ROM GETIN (PUSH) * ROMGET
198 ROM GETIN (PUSH) * ROMGET
199 ROM GETIN (PUSH) * ROMGET
200 ROM GETIN (PUSH) * ROMGET

```

## EXTENDING BASIC (continued from p.38)

routine, because it isn't a subroutine ending with an RTS, as we will see in a minute.

First the routine tries to see if there's enough space on the stack to do a GOSUB. The routine needs three bytes space on the stack. This value is loaded into the accumulator (line 1189) and then the test stuff is made (line 1194). If there isn't enough space left on the stack, routine SA11B ends with the dreaded error report "OUT OF MEMORY".

Otherwise, five bytes are pushed onto the stack, the first two bytes point at the parameter after the GOSUB command. Remember, SA11B are part of the CHARGET routine and we were, point to the next character. This is done by pushing it onto the stack (line 1200-1204).

Next, the current line number which is contained in zero page 104 (A) is pushed onto the stack (line 1248-1250). Finally, the token value for GOSUB (1800) is pushed onto the stack (line 1260-1264).

Whenever a RETURN command is given, the interpreter searches for a

stack of data like this, starting with 1800. Then the stack pointer is modified so that it points to the block, and the block is pulled back from the stack. Now the Basic program flow can jump back to the proper line, and the character pointer in SA11B will again point at the parameter as that line. All that has to be done now is find the beginning of the sub routine so that program execution can continue as from there.

In line 1300 of our routine we call the GOTO routine, which makes the Basic program jump to the subprogram which has been called. Finally (line 1304), we jump to the ROM routine at SA1AE which executes the next command. Please note that we jump to the routine as opposed to not returning to it as with the other routines. Can you guess why?

In lines 1280-1290 of our GOSUB routine we have pushed five bytes onto the stack. This changes the stack pointer by five locations! Therefore it would be impossible to use an RTS to return us to the proper routine as line 480.

In the next article, I'd like to expand on GOTO and GOSUB by developing a labeling system for Basic which works in the same way most assembler programs. That is, you can define a label and then use GOTO and GOSUB to jump to the line commented as that label.

## Figure 1: ROM Routines Used

- SA1DA Evaluate an expression and put the result into Y register
- SA1DT Convert value in Y register to 16-bit integer and put result into zero page SA1A15 and Y register accumulator
- SA1A3 Execute GOTO line number contained in SA1A15
- SA1FB Test to see if there's enough room on the stack to push the number of bytes onto the stack which is contained in the accumulator. If not, give error report "OUT OF MEMORY"




All advertisements in this section must be prepaid. Advertisements are accepted subject to the terms and conditions available on the advertisement rate card available on request.



Send your requirements to  
ANNA EDDIE  
CLASSIFIED DEPARTMENT  
ASP LTD 1 GOLDEN SQUARE  
LONDON W1R 3AB

**SPECIAL  
OFFER**

2000-2001  
 2001-2002  
 2002-2003  
 2003-2004  
 2004-2005  
 2005-2006  
 2006-2007  
 2007-2008  
 2008-2009  
 2009-2010  
 2010-2011  
 2011-2012  
 2012-2013  
 2013-2014  
 2014-2015  
 2015-2016  
 2016-2017  
 2017-2018  
 2018-2019  
 2019-2020  
 2020-2021  
 2021-2022  
 2022-2023  
 2023-2024  
 2024-2025  
 2025-2026  
 2026-2027  
 2027-2028  
 2028-2029  
 2029-2030  
 2030-2031  
 2031-2032  
 2032-2033  
 2033-2034  
 2034-2035  
 2035-2036  
 2036-2037  
 2037-2038  
 2038-2039  
 2039-2040  
 2040-2041  
 2041-2042  
 2042-2043  
 2043-2044  
 2044-2045  
 2045-2046  
 2046-2047  
 2047-2048  
 2048-2049  
 2049-2050  
 2050-2051  
 2051-2052  
 2052-2053  
 2053-2054  
 2054-2055  
 2055-2056  
 2056-2057  
 2057-2058  
 2058-2059  
 2059-2060  
 2060-2061  
 2061-2062  
 2062-2063  
 2063-2064  
 2064-2065  
 2065-2066  
 2066-2067  
 2067-2068  
 2068-2069  
 2069-2070  
 2070-2071  
 2071-2072  
 2072-2073  
 2073-2074  
 2074-2075  
 2075-2076  
 2076-2077  
 2077-2078  
 2078-2079  
 2079-2080  
 2080-2081  
 2081-2082  
 2082-2083  
 2083-2084  
 2084-2085  
 2085-2086  
 2086-2087  
 2087-2088  
 2088-2089  
 2089-2090  
 2090-2091  
 2091-2092  
 2092-2093  
 2093-2094  
 2094-2095  
 2095-2096  
 2096-2097  
 2097-2098  
 2098-2099  
 2099-2100  
 2100-2101  
 2101-2102  
 2102-2103  
 2103-2104  
 2104-2105  
 2105-2106  
 2106-2107  
 2107-2108  
 2108-2109  
 2109-2110  
 2110-2111  
 2111-2112  
 2112-2113  
 2113-2114  
 2114-2115  
 2115-2116  
 2116-2117  
 2117-2118  
 2118-2119  
 2119-2120  
 2120-2121  
 2121-2122  
 2122-2123  
 2123-2124  
 2124-2125  
 2125-2126  
 2126-2127  
 2127-2128  
 2128-2129  
 2129-2130  
 2130-2131  
 2131-2132  
 2132-2133  
 2133-2134  
 2134-2135  
 2135-2136  
 2136-2137  
 2137-2138  
 2138-2139  
 2139-2140  
 2140-2141  
 2141-2142  
 2142-2143  
 2143-2144  
 2144-2145  
 2145-2146  
 2146-2147  
 2147-2148  
 2148-2149  
 2149-2150  
 2150-2151  
 2151-2152  
 2152-2153  
 2153-2154  
 2154-2155  
 2155-2156  
 2156-2157  
 2157-2158  
 2158-2159  
 2159-2160  
 2160-2161  
 2161-2162  
 2162-2163  
 2163-2164  
 2164-2165  
 2165-2166  
 2166-2167  
 2167-2168  
 2168-2169  
 2169-2170  
 2170-2171  
 2171-2172  
 2172-2173  
 2173-2174  
 2174-2175  
 2175-2176  
 2176-2177  
 2177-2178  
 2178-2179  
 2179-2180  
 2180-2181  
 2181-2182  
 2182-2183  
 2183-2184  
 2184-2185  
 2185-2186  
 2186-2187  
 2187-2188  
 2188-2189  
 2189-2190  
 2190-2191  
 2191-2192  
 2192-2193  
 2193-2194  
 2194-2195  
 2195-2196  
 2196-2197  
 2197-2198  
 2198-2199  
 2199-2200  
 2200-2201  
 2201-2202  
 2202-2203  
 2203-2204  
 2204-2205  
 2205-2206  
 2206-2207  
 2207-2208  
 2208-2209  
 2209-2210  
 2210-2211  
 2211-2212  
 2212-2213  
 2213-2214  
 2214-2215  
 2215-2216  
 2216-2217  
 2217-2218  
 2218-2219  
 2219-2220  
 2220-2221  
 2221-2222  
 2222-2223  
 2223-2224  
 2224-2225  
 2225-2226  
 2226-2227  
 2227-2228  
 2228-2229  
 2229-2230  
 2230-2231  
 2231-2232  
 2232-2233  
 2233-2234  
 2234-2235  
 2235-2236  
 2236-2237  
 2237-2238  
 2238-2239  
 2239-2240  
 2240-2241  
 2241-2242  
 2242-2243  
 2243-2244  
 2244-2245  
 2245-2246  
 2246-2247  
 2247-2248  
 2248-2249  
 2249-2250  
 2250-2251  
 2251-2252  
 2252-2253  
 2253-2254  
 2254-2255  
 2255-2256  
 2256-2257  
 2257-2258  
 2258-2259  
 2259-2260  
 2260-2261  
 2261-2262  
 2262-2263  
 2263-2264  
 2264-2265  
 2265-2266  
 2266-2267  
 2267-2268  
 2268-2269  
 2269-2270  
 2270-2271  
 2271-2272  
 2272-2273  
 2273-2274  
 2274-2275  
 2275-2276  
 2276-2277  
 2277-2278  
 2278-2279  
 2279-2280  
 2280-2281  
 2281-2282  
 2282-2283  
 2283-2284  
 2284-2285  
 2285-2286  
 2286-2287  
 2287-2288  
 2288-2289  
 2289-2290  
 2290-2291  
 2291-2292  
 229

[illegible]

## 01-437 0000

59. **NOT AN ANSWER**

We can also supply hardware, software, stand alone accessories and spare parts.  
 Call Dennis Pitt for more info 01454 751111

# Repairs Guide CALL

ANNA EDDIE

ON

01-437 0699

## REPAIRS

AMIGA ATARI etc.  
MONITORS & ALL ADD ON's  
AMIGA UPGRADES AND MOD's

All estimates FREE

Prices from only £19.90

Call or post FAST TURNAROUND

PVS ELECTRONIC COMPONENTS

244 Deansgate Court,

Deansgate, Manchester M3 4HQ

Tel 061 831 7088 Telex 888488

Fax 061 432 6934



### COMPUTER KANUT T

REPAIRS  
*Readcare*  
THE COMMODORE SPECIALIST

REPAIRS  
FLIGHTS  
ON-SCREEN  
AMIGA

FAST  
EFFICIENT  
SERVICE

FOR REPAIRS CALL

01 251 2832

(N.W. LONDON-HARROW)

### COMMODORE SPARES & REPAIRS

REPAIRS  
FLIGHTS  
ON-SCREEN  
AMIGA  
REPAIRS  
FLIGHTS  
ON-SCREEN  
AMIGA  
REPAIRS  
FLIGHTS  
ON-SCREEN  
AMIGA



Unit 40, 14 BRISTOL ROAD, THURROFT  
SILFORD, CHESHIRE, WY 8 7JL. (0925) 444411

THE TELEPHONE NUMBER  
AT THE TOP OF THIS  
PAGE IS FOR  
ADVERTISERS ONLY  
ANNA EDDIE IS UNABLE  
TO ISSUE DETAILS OF  
REPAIR CENTRES OTHER  
THAN THOSE  
ADVERTISED HERE.

### SPARES REPAIRS SALES

all makes of computers & peripherals  
Free Reliable Guaranteed Service

Contact Dave Barber at long. 448828

### 44 ELECTRONIC SERVICES

88 Woodgate Avenue, St. Beddles,  
Chesham, Bucks. HP11 1AD  
Tel 0296 - 248874

### COMPUTER REPAIRS (UK)

Specialist Repairs To All

Compatible Equipment

\* Free survey, £25. VAT

\* All work to be guaranteed

\* Delivery & collect on arrival

\* 4.950 hrs. available

\* No other equipment or brand restrictions

\* Insurance

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

\* 24 hours service

### CRUYDON COMPUTER CENTRE

24 Edgware Road, Tottenham, North

London, N17 9JL Tel 0181 244 3448

COMPUTER SERVICE

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

### VTS COMPUTER REPAIRS

01-407 8625

Simply send your unit to the address below  
security packed with full description and  
photos

COMMODORE, AMIGA, ATARI, SAMSUNG  
(UK)

### VTS COMPUTER REPAIRS

12 Lytle House, Ch. Deane, Warrington

Cheshire, W14 4DD

FAST, RELIABLE TURNAROUND

### COMMODORE SERVICE CENTRE

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

### COMMODORE REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

REPAIRS

### Computer Repairs Fixed Super Low Prices!

without use of parts, labour and VAT



FAST AS  
FOR  
TECHNOLOGY

1 YEAR WARRANTY - Collectors/delivery available for local areas

### \* SUPER OFFERS \*

SPRINT/AM	£14.95	Free Screen	ELECTRON	£22.95
512K	£14.95	Free Screen	COMMODORE 128	£22.95
512K	£14.95	Free Screen	COMMODORE 250	£22.95
512K	£14.95	Free Screen	COMMODORE 400	£22.95
512K	£14.95	Free Screen	COMMODORE 640	£22.95
512K	£14.95	Free Screen	COMMODORE 800	£22.95
512K	£14.95	Free Screen	COMMODORE 1000	£22.95
512K	£14.95	Free Screen	COMMODORE 1200	£22.95
512K	£14.95	Free Screen	COMMODORE 1400	£22.95
512K	£14.95	Free Screen	COMMODORE 1600	£22.95
512K	£14.95	Free Screen	COMMODORE 1800	£22.95
512K	£14.95	Free Screen	COMMODORE 2000	£22.95
512K	£14.95	Free Screen	COMMODORE 2200	£22.95
512K	£14.95	Free Screen	COMMODORE 2400	£22.95
512K	£14.95	Free Screen	COMMODORE 2600	£22.95
512K	£14.95	Free Screen	COMMODORE 2800	£22.95
512K	£14.95	Free Screen	COMMODORE 3000	£22.95
512K	£14.95	Free Screen	COMMODORE 3200	£22.95
512K	£14.95	Free Screen	COMMODORE 3400	£22.95
512K	£14.95	Free Screen	COMMODORE 3600	£22.95
512K	£14.95	Free Screen	COMMODORE 3800	£22.95
512K	£14.95	Free Screen	COMMODORE 4000	£22.95
512K	£14.95	Free Screen	COMMODORE 4200	£22.95
512K	£14.95	Free Screen	COMMODORE 4400	£22.95
512K	£14.95	Free Screen	COMMODORE 4600	£22.95
512K	£14.95	Free Screen	COMMODORE 4800	£22.95
512K	£14.95	Free Screen	COMMODORE 5000	£22.95
512K	£14.95	Free Screen	COMMODORE 5200	£22.95
512K	£14.95	Free Screen	COMMODORE 5400	£22.95
512K	£14.95	Free Screen	COMMODORE 5600	£22.95
512K	£14.95	Free Screen	COMMODORE 5800	£22.95
512K	£14.95	Free Screen	COMMODORE 6000	£22.95
512K	£14.95	Free Screen	COMMODORE 6200	£22.95
512K	£14.95	Free Screen	COMMODORE 6400	£22.95
512K	£14.95	Free Screen	COMMODORE 6600	£22.95
512K	£14.95	Free Screen	COMMODORE 6800	£22.95
512K	£14.95	Free Screen	COMMODORE 7000	£22.95
512K	£14.95	Free Screen	COMMODORE 7200	£22.95
512K	£14.95	Free Screen	COMMODORE 7400	£22.95
512K	£14.95	Free Screen	COMMODORE 7600	£22.95
512K	£14.95	Free Screen	COMMODORE 7800	£22.95
512K	£14.95	Free Screen	COMMODORE 8000	£22.95
512K	£14.95	Free Screen	COMMODORE 8200	£22.95
512K	£14.95	Free Screen	COMMODORE 8400	£22.95
512K	£14.95	Free Screen	COMMODORE 8600	£22.95
512K	£14.95	Free Screen	COMMODORE 8800	£22.95
512K	£14.95	Free Screen	COMMODORE 9000	£22.95
512K	£14.95	Free Screen	COMMODORE 9200	£22.95
512K	£14.95	Free Screen	COMMODORE 9400	£22.95
512K	£14.95	Free Screen	COMMODORE 9600	£22.95
512K	£14.95	Free Screen	COMMODORE 9800	£22.95
512K	£14.95	Free Screen	COMMODORE 10000	£22.95

Please confirm payment by cheque or credit card only. Please confirm order with us by 1

### W.T.S. ELECTRONICS

101 Station Master House, Chesham, Bucks, HP11 1AD. Tel 0296 499444

acceptance

YOUR COMMODORE, March 1989

B

R

E

A

H

## Readers Problems

Though the Commodore fills a role of the world's most popular microcomputer, it can be very difficult to find specific information about your particular machine.

At the Four Commodore office we receive literally hundreds of letters from you, our readers, on a wide range of subjects ranging from the simple "Can you give me the telephone number for..." to the more complex "I'm trying to write a program that does a split screen. How do I do it?"

Unfortunately, the volume of mail received has become so great that it is impossible to answer every letter and still manage to publish a magazine each month.

For this reason we have felt it necessary to produce a number of guidelines for getting information from us:

- 1) We cannot guarantee to answer every letter that is to the magazine. Should it become apparent that a number of readers are suffering from the same problem, then we will reply by the letter via the Letters page.
- 2) A new telephone has been set up. This will be open for your queries on

Tuesday and Thursday afternoons between 3.00pm and 4.00pm. We will not be able to deal with your telephone queries at any other time. If our technical adviser is not available when you ring, then a message will be taken.

3) If you are having problems with one of our listings, can you please let us know as writing. This will enable us to see if a number of people are having the same problem. When a common problem becomes apparent with a program, then a correction sheet will be issued. Enclose a self-addressed, stamped envelope and we will send you a copy of the correction sheet as soon as it is available.

We are very that it has become necessary to arrange these rules. However, we are sure that you will agree with us that the more time that we can spend making Your Commodore the most informative magazine around the better.

For program queries write to:

Program Comments

Your Commodore

1 Golden Square

London

W1R 3AB

If you wish to telephone then call:

01-407 0828 Extn 212

## Commodore Where Are You?

At the 1 to 4 Commodore office we are regularly asked for the address and telephone number of Commodore UK. Many people after referring to their computer manuals, believe that it is based in Cardiff.

The Commodore plant in Cardiff is closed down some time ago. Regrettably here you will find the correct address for Commodore UK.

We repeat that you want the correct address to the front of your

computer manual for future reference.

Commodore Business Machines (UK)

Commodore House

The Southbank

Cardiff Road

Manchester

M16 7YA

Tel 06251 770085

## William Tell

If you have been typing in the William Tell program over the last few months you will be pleased to note that it is completed in this issue.

In order to get the program to run correctly you should follow the procedure below:

- 1) Make sure that you have the 4 file on loader to hand.
- 2) If using cassette make sure that you have a new one handy and that it is a long one.
- 3) Type in and SAVE the program below onto your disk or onto the cassette mentioned in 1 above. If using cassette change the J's to I's.
- 4) Turn OFF and ON your machine.
- 5) LOAD the first Basic loader and RUN it. A new program will be saved out once this is finished. If using cassette make sure that you place the new cassette into the recorder before you RUN the loader. The new program should be saved directly after the program you have just entered in 5.
- 6) Turn OFF and ON your machine.
- 7) LOAD the next Basic loader and repeat the steps 5 to 7 for all remaining Basic loaders.
- 8) To start the William Tell program simply LOAD and RUN the first program that you SAVEd. This will then LOAD and RUN the rest of the game.

## PROGRAM WILLIAM TELL

```
10 GOTO=HENDLOAD/WT/ J:J A=J+1
20 IF=HENDLOAD/WT/ J:J A=J+1
30 GOTO=HENDLOAD/WT/ J:J A=J+1
40 IF=HENDLOAD/WT/ J:J A=J+1
50 IF=HENDLOAD/WT/ J:J A=J+1
60 IF=HENDLOAD/WT/ J:J A=J+1
70 GOTO=HENDLOAD/WT/ J:J A=J+1
80 GOTO=HENDLOAD/WT/ J:J A=J+1
```

## The Nibbles By Alan Batchelor



AMIGA

ATARI  
ST

IBM PC

APPLE  
MAC

THE SPECIALIST  
MUSIC MAGAZINE  
FOR COMPUTER USERS

MUSIC X-The Review

MICRO MUSIC

Find out about the magazine  
the computer music world has been  
waiting for

KCS-Just what the  
Doctor ordered!!

ACORN

COMMODORE  
64

AMSTRAD  
CPC

ZX

SPECTRUM

# THE TIME HAS COME...

Britain's First Music Magazine for the Computer User!

Featuring Reviews on Hardware & Software across all formats - Micro Music is the magazine the Computer Musician has been waiting for.

And look for it in all good Newsagents.

# Emlyn Hughes



## INTERNATIONAL



RECOMMENDED BY  
**MATCH**

Quite simply the best football  
simulation available for the 64.  
Nothing short of superb.

**ZZAP 64**

**AVAILABLE NOW FOR  
COMMODORE 64**  
**£9.95 tape £12.95 disk**  
Coming soon for Spectrum & Amstrad

## Audiogenic

Audiogenic Software Limited, Winchester House,  
Canning Road, Harrow HA3 7SU, England

Order by phone on 01 861 1166



# Hughes



## INTERNATIONAL



# Audiogenic

Audiogenic Software Limited, Winchester House,  
Canning Road, Harrow HA3 7LJ England

Order by phone on 01 861 1166



# YOUR COMMODORE

## COMMODORE GRAPHICS THE INSIDE STORY

GRAPHICS GENERATED  
THROUGH FLUKE  
A REVOLUTIONARY  
VIDEO SYSTEM  
WITH  
TWO BIT TECH



UNDETECTABLE

ENTER YOUR BASIC BOX